

MOJ MIKRO

beležnik 1000 / br. 3 / godišina 7 / izdanje 40 strani

PRILOG

Objekti u računarskoj grafici
(Turbo Pascal 5.5)

XWINDOW

Standard budućnosti?

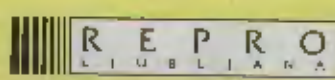


FOXPRO
Baza podataka sa pedigreeom
PREKUPOVINE RAČUNARA (3)
Lego za hakere
AT EMULATORI ZA ATARI ST

ŽIVETI SA MONITOROM
Da li je rad računarima
bezbedan?

 **AUTODESK**
AUTOCAD®
RELEASE 11

d.o.o.
CELOVŠKA 175-YU
61107 LJUBLJANA
TELEFON 061/552-341, 552-150, 554-450 FAX (061) 552-583.
TLX 31 639 yu-autena p.p. 69



RECOGNITA PLUS
Madžari
čitaju brže

YU ISSN 0352-6054



9 770352 605000

Izlazi u dva izdanja: slovenačkom i srpskohrvatskom

SADRŽAJ

Hardver

Pre kupovine računara (3)	11
AT emulatori za Atari ST	20

Softver

XWindow - standard budućnosti?	8
FoxPro: baza podataka sa pedigreeom	22
Recognita Plus: Madžari čitaju brže	24

Zanimljivosti

Da li je rad računarom bezbedan?	12
Uvoz programa: nikada poštom	16
Kritičari: Zaključaj podatke javnim ključem	53

PRILOG

Objekti u računarskoj grafici (Turbo Pascal 5.5)	43
--	----

Rubrike

Mimo ekrana	6
Mali oglasi	68
Hitna pomoć	68
Recenzije	67
Zabavni matematički sadaci	66
Vaš mikro	58
Igre	68



Strana 12:
Koliko je bezbedan rad računarima?



Strana 22:
FoxPro: brz kao lisica.

Strana 59: Indianapolis 500 i druge igre.



Jesmo li zastupnici ili nismo? Dnevni list Delo je 20. decembra prošle godine preko jedne šestine strane objavio neobičan oglas. U njemu je Sun Microsystems, Inc., International Operations saopštavao da su njegovi jedini zastupnici u Jugoslaviji ljubljanski Mikrohit i beogradski Aeroinženjering. Pored toga, uz dve odbojne ilustracije, upozoravao je kupca da ne nabavljaju njegovu opremu uvedenu preko drugih, nama nepoznatih dobavljača (npr. Graf, Sledi, Cardem itd.). Sledećeg dana, preduzeće Graf, d. o. o., na istom je mestu u Delu odgovorilo oglasom iste visičine i na kraju zapretilo: Ovim saopštenjem nanosi se velike štete i principalu Sun Microsystems, koji je upoznao sa jučerašnjim saopštenjem i uskoro se može očekivati njegova zvanična reakcija.

Tako je javnost po prvi puta upoznata sa jednom od ružnih priča koje kolaju od štanda do štanda na YU računarskim sajamovima. Jedan zastupnik nam se požalio da je njegov ovlašćeni prodavac na svoju ruku nabavljao programe u Italiji. Jedan drugi zastupnik, kako veći konkurencija, već se pet godine snabdeva kod nekog Švajcarca, koji ima dozvolu za prodaju samo na domaćem tržištu...

Jesmo li hajduci ili nismo? Poslednjih nekoliko sedmica, slovenačke računardžije se smeju sledećoj anegdoti. Zastupnik svetski poznate softverske kuće prodao je dvema jugoslovenski poznatim firmama iz Ljubljane po jedan (1) primarak programa. Zatim se neko setio biblijske priče o umnožavanju hlebova i riba, i jedeše svi, i nasitiše se... (Matija 14, 20.) Programi sada rade na dvesta računara. Gladni zastupnik ne produzima ništa jer zvanično još ništa ne zna. A AZIL - asocijacija za intelektualno vlasništvo - privremeno je zamenio. Njegov telefonski broj je fata morgana, rekli su nam saradnici ili, programa Hrvatskog radija, koji su pripremili okrugli sto o zaštiti softvera kod nas.

Jesmo li Belkanci ili nismo? YU računarsko tržište ponajviše podseća na srednjovekovni sajam sa mešetarima i žonglerima, klovnovima i mečkama. Umesto da uvedu red u svojim redovima, zastupnici se smučaju oko naše redakcije i na sve moguće načine pokušavaju da sami ocnaju programe koje prodaju. Jedan poznati privatnik bio je spreman da napiše članak pod pseudonimom, zatim se zaklinjao da nećemo dobiti program na recenziju, a na kraju je (uzalud) zvao u pomoć naše komercijalno odeljenje. Na drugoj strani, čak trojica žonglera koji se izdaju za naše saradnike, istovremeno moljakuju nekog američkog izdavača za jedan te isti program.

Moj mikro nije ni klovn ni mečka. Zato sa YU zastupnicima stranih računarskih firmi nećemo još dugo da pregovaramo. A sa principanima razgovaraćemo sami. Zastupnik Microsofta za istočnu Evropu već je posetio sve naše računarske časopise i predložio im saradnju bez posrednika.

Prosto po Rolling Stones: I know it's only rock 'n' Tetrís, but I like it.

Glavni i odgovorni urednik časopisa Moj mikro ALJOŠA VREČAR • Zamenik glavnog i odgovornog urednika SLOBODAN VUJANOVIĆ • Dizajner i tehnički urednik ANDREJ MAVŠAR • Sekretarica ELICA POTOČNIK • Stručni savetnik MATEVŽ KMET, dipl. Ing.

Izdavački savet: Ajenka MIŠIČ (Gospodarska zbornica Slovenije), predsednika, Ciril BEZLAJ (Gorenje - Proceana oprema, Titova Velanje), prof. dr Ivan BRATKO (Fakulteta za elektrotehniku, Ljubljana), prof. Aleksander GOKAN (Državna založba Slovenije, Ljubljana), Borislav RADŽIBABIĆ, dipl. Ing. (Energoprojekt, Energo-Data, Beograd), dipl. Ing. Miloš KOBE (Istra, Ljubljana), mag. Ivan GERLIČ (Zveza organizacij za tehničko kulturu, Ljubljana), Tone POLENEC (Mladinsko knjiga, Ljubljana), dr. Marjan ŠPEGEL (Inštitut Inženj. Stefan, Ljubljana), Zoran STRBAC (Mikrohit, Ljubljana).

Moj mikro izdaje DELO - REVUE, p.o. Titova 35, 61001 Ljubljana • štampa: Delo - TČR, grafična dejavnost, p.o., Titova 35, Ljubljana. Direktor Delo - Revue ANDREJ LESJAK • Nemanjanih materijala ne vraćamo. • Na osnovu mišljenja Republičkog komiteta za informiranje br. 421-1/72, od 25. V. 1984, Moj mikro oslobođen je posebnog paraža na promet.

Adresa redakcija: Moj mikro, Titova 35, 61001 Ljubljana, telefon (061) 315-066, 319-796, telefax: 311-871, 319-280, telex: 31-255 YU DELO

Oglasno izdanje: France Logonder, tel. (061) 315-366, int. 27-14, Delo - STIK, Titova 35, 61001 Ljubljana, tel. (061) 315-366, int. 28-95.

Prodaja i pretplata: Delo - Prodaja, p.o., 61001 Titova 35, Ljubljana; kolportaza: telefon: (061) 319-780; pretplata: telefon: (061) 319-255, 318-265 i 315-366, lokal 21-53. Uplatnica za plaćanje pretplate šalju se na tri puta godišnje.

Godišnje pretplate za inostranstvo: 458 ATS, 44.800 ITL, 50 DEM, 50 CHF, 204 FRF, 35 USD.

Uplate na žiro račun: DELO - REVUE, Ljubljana (za Moj mikro) 50102-603-48914.

Uplate na devizni račun pe LB 36, Ljubljana: 50100-620-107-257300-27821/1.

UNICO

d.o.o., Cankarjeva 4, Ljubljana, tel. (061) 221-838

distributer **MICROSOFT INC.** za Jugoslaviju

distributer Peter Norton za Jugoslaviju

zastupanje Fox Software i STSC-Statigraphics



IBM PS/2 model 90

IBM je, iz svoje familije računara, izbacio na tržište nov model PS/2 i sa njime prilično iznenadilo kompjuterski svet. Računar je izgrađen oko procesora 80486/33, ima 8 Mb RAM, 160 Mb disk SCSI, a najviše iznenađuje novi standard za grafiku - IBM ga naziva XGA (extended graphics array) koji omogućava rezoluciju 1024 x 768 u 256 boja.

Prva stvar koja se kod Modela 90 primjećuje je beskonačno dugo vreme startovanja sistema: skoro minut i po. Jedan od razloga je taj, da se na startu izvodi ceo niz programa POST (power on self test) i da računar ima ogromnu memoriju. Još i to, većina BIOS-a se ubacuje iz tvrdog diska umesto iz ROM-a.

Grubo gledano, model 90 je opet napravljen po IBM-ovoj konzervativnoj logici i, premda dobar, to je nedorađen računar. Model 90 je, sa cenom od 8500 GBP

za verziju od 33 MHz i tvrdim diskom od 160 Mb i od 9540 GBP za verziju od 33 MHz, daleko od jeftinog računara. Ali, to smo od IBM i očekivali. IBM je, suprotno ostalim proizvođačima računara, otkrilo da korisnici žele računar za veoma širok spektar upotrebe. Model 90 je sposoban da radi na donjoj ivici tržišta servera podataka, a istovremeno je na vrhu tržišta stonih računara PC (PS/2). Razumljivo da je IBM digao mnogo prašine oko arhitekture MC i tehnički još uvijek ostaje jedan od vodećih proizvođača mikrokanala. U praksi je IBM-ova filozofija teko konzervativna da će standard AT ISA uskoro dostići sve kvalitete koje MC nudi. EISA će IBM jednostavno pregaziti.

Ako ste vezani na računara IBM, model 90 je ono, što ste očekivali dugo godina. Ali ako vam je za ime svejedno, na tržištu postoji tuco brzih računara sa dobrom super VGA grafikom koji rade sve što radi IBM PS/2, model 90 i to jeftinije i brže.

Ekranški filter štiti od niskofrekventnog zračenja

Svi koji su bavili kompjuterima, znate da monitori emituju opasno zračenje. Preduzeće NoRad iz Kalifornije je za 129 USD napravilo filter za ekran, pod nazivom Shield (štiti), koji zaustavlja zračenje. Dok se neki još pitaju da li nes monitor postepeno ubija ili ne, sigurno je da Shield štiti od nekih niskofrekventnih signala koje monitor emituje i istovremeno smanjuje neprijatan osetaj površine ekrana.

Kao i mnogi drugi filteri i Shield je napravljen od fine tamne mrežice, ali ima dve važne novosti: mrežica je mnogo finija, praktično nevidljiva (osim kad se njena površina detaljno pogleda) i istovremeno konduktivna. Sve vlasti mrežice su kao napako izokrenute žice, na jezgru od izolacionog materijala navljen je sloj bakra i nikla, a vlasti su presvučene crnim mat metalnim fibrom koji smanjuje osetaj monitora. Mrežica ima minimalan uticaj na jasnost slike, a istovremeno efikasno sprečava osetaj.

Shield se pričvršćuje plastičnom kućište monitora. NoRad uz Shield daje i tri samolepljive trake, kojima se filter fiksira u proizvoljan položaj, i sprej za čišćenje ekrana pre postavljanja filtera.

Da bi Shield odvodilo radijaciju, treba ga uzemljiti. U tu svrhu treba ga, žicom dužine 90 cm, povezati jednim od konektora na zadnjoj strani računara. Uzemljeni Shield štiti od niskofrekventnog zračenja električnog polja u području iznad 20 KHz.

NoRad nudi različite veličine i modele filtera Shield, sve do onih sa dijagonalom od 19 palaca. Između ostalih postoje i "meki" filteri koji se savijaju i prilagođavaju zaokruženom obliku monitora.

Shield ne zaustavlja ELF (vanredno niskofrekventno zračenje pod 300 Hz) i bočno zračenje. Pretpostavlja se da je zračenje ELF kancerogeno, ali trenutno ga ne zaustavlja nijedan filter ili monitor (ili televizor). Ako se bojite (isto tako opasnog) visokofrekventnog zračenja, statičko-električnog polja i slabog kontrasta, Shield će vas uspešno zaštititi. Jedina žrtva je minimalno lošija oštrina slike.

Cenovnik: cena za filtere od 8,38 x 7,38 do 12,38 x 9,5 col je 129 USD, za dodatak za prilagođavanje filtera 10 USD, a za 19-palačne ekrane je 349 USD. NoRad Corp., 1649 11th St., Santa Monica, CA 90401.

Program AUTOCON za upravljanje AUTOEXEC/CONFIG

Većina ozbiljnih korisnika PC, manje više puta svoje pogonske datoteke AUTOEXEC.BAT i CONFIG.SYS. Ako imate ekvivalent ili sistemske programe i pogonske uređaje koje želite da svaki put unosite, verovatno ste već napravili više verzija pogonskih datoteka AUTOCON, program u javnoj distribuciji (shareware) omogućava za samo 18 USD da napravite i koristite bazu podataka u kombinaciji AUTOEXEC/CONFIG, tako da konfiguraciju možete brzo i jednostavno promeniti.

AUTOCON smešta do 50 različitih zapisa, gde je svaki zapis jedna od kombinacija AUTOEXEC/CONFIG. Listati možete kroz datoteke i aktivirati sistem proizvoljnom verzijom pogonskih datoteka. Svaki zapis ima svoje ime. Računar se, na primer, aktivira kombinacijom WIN3 koja je prilagođena za Microsoftove Windows 3.0, tako da u DOS ispisete AUTOCON WIN3.



AUTOCON, pri ponovnom aktiviranju, zamenjuje trenutne datoteke AUTOEXEC.BAT i CONFIG.SYS sa onima koje se nalaze pod navedenim imenom i resetira računar. Ugrađeni tekst editor omogućava da sve popravke u pogonskoj datoteci napravite u AUTOCON-u. Tekst editor koristi WordStarove naredbe i onta je u redu, a naredbe možete i prilagoditi.

Ako za disk koristite predmemoriju, možete narediti program AUTOCON da pre resetiranja isprazni predmemorijski fond. To je neophodno kada predmemorijski program koristi kašnjenje upravljanja na disk, jer bi inače sve promene koje AUTOCON napravi pre resetiranja na disk upisale na disk.

AUTOCON je jednostavno i pristupačno rešenje za podešavanje različitih pogonskih konfiguracija. Slični programi na tržištu su CONFIG.CTL i BOOT.SYS, ali nisu tako prijatni.

AUTOCON, ver. 1.3a, cena 18 USD, zahteva: 265 K RAM, DOS 2.0 ili viši, Larry Weaver, P.O. Box 2639, Wenatchee, WA 98093-2639.

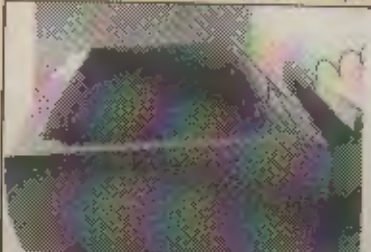
Telefaks za diskove

Često razmišljamo kako bi dobili »jednostavnu ideju«, neku koja bi se neopto- vetno lako razvila i realizovala, a i da je jeftina. Ali kolikogod je ideja jednostavna, veći utisak ostavlja na većinu kupaca koji obično misle: »Uh, mogao sam toga sam da se setim!«

Telefaks je dobar primer jednostavne ideje. Kada je svet još slao originalna dokumenta preko pošte, a sve razgovore obavljao preko telefona, neko se setio i napravio telefaks koji šalje slike preko telefona.

Baš tako su došli do jednostavne ideje u Alfa Systems-u gde su razvili DiskFax.

To je aparat koji radi isto kao i telefaks (košta približno toliko), ali šalje i prima podatke sa diskete. Ideja bazira na jednostavnoj činjenici da korisnici ne vole da se muče sa modemima, jer je njihov



potrebno izvesno znanje, softver i šta ja znam šta sve još. Ali, može li se disketu jednostavno ubaciti u pogon, reći koje datoteke da se prenese i kuda i samo pritenuti na dugme? DiskFax baš to radi. To je samostalna kutija veličine računara mini-tower koja na prednjoj strani ima dva disketna pogona (5,25 i 3,5 palaca), a na vrhu telefonski brojčanik i tri tastera.

Postupak je jednostavan: pritisne se dugme sa oznakom SEND, ubaci se disketa u pogon, pričekajte da DiskFax prepozna disketu, označe se datoteke koje treba poslati, odkuca telefonski broj i stvar je završena. DiskFax vreme slanja, prikazuje na ekranu od tačnih kristala koliko procenata datoteke je već otposlao.

DiskFax prepoznaje diskete DOS i UNIX, a uskoro će biti na raspolaganju i prinos Appleovih disketa. Brzina prenosa je 9600 bps sa kompresijom i kontrolnom greškom izrađena je i jača verzija DiskFaksa sa ugrađenim 20 Mb tvrdim diskom. U tom slučaju jedinica se može ostaviti upaljena i sve pripremljene datoteke skupljaju se na disku zajedno sa informacijama o njihovim formatima, veličinama, pošiljaocu i vremenu prijepa. Kada treba poslati datoteku, ubacite disketu u pogon označite na ekranu željenu datoteku i pritisnite na dugme UNLOAD.

Za sada DiskFax prodaju samo u Engleskoj, a cena mu je 800 GBP (1000 GBP za jedinicu sa tvrdim diskom).

Kartica Expanz! umesto većeg tvrdog diska



InfoChip System izradilo je slot karticu Expanz! koja u realnom vremenu komprimira i dekomprimira podatke na disku i radi za cenu od 199 USD alternativu za veći tvrdi disk koji inače košta više stotina dolara. Kada se kartica stavi u bilo koju 8 ili 16-bitnu utičnicu na matičnoj ploči računara, Expanz! nevidljivo komprimira sve podatke koje treba snimiti na disk i dekomprimira ih kada se oni vraćaju u memoriju.

Kada se disk inicializira i radi pod karticom Expanz! (postupkom se istovremeno komprimiraju svi podaci koji su bili na disku), svaka datoteka na disku zauzima samo desetinu prostora kojeg bi zauzela na običnom disku DOS. Komprimirani se mogu i svi programi na disku sa kojima se sistem stavlja u pogon.

Sve potrebne programske podrške je kratak pogon koji se kroz CONFIG.SYS automatski unosi u memoriju, a radi samo do verzije DOS 5.3D. Za ovu godinu su obećani još pogon za DOS 3.31, DOS 4.0, DS/2 i za funkcionisanje Expanz! u mreži.

Expanz! komprimiranje i dekomprimiranje podataka vrši pomoću čipova na kartici. Komprimiranje vrši ugrađeni algoritam u procesor IC-105, a proces je

tako brz da razliku verovatno nikada nećete primetiti. Za poređenje: jednak softverski algoritam prouzrokuje za komprimiranje veoma primetno usporavanje protoka podataka.

Cena koju treba platiti za komprimiranje sa karticom Expanz! je gubitak 30 K RAM kojeg potroše pogoni i 16 K RAM koji se upotrebljava kao memorijski prozor. Nažalost, pri korišćenju kartice Expanz! otpadaju sva oruđa za zaštitu i ubravanje diska. Kada disk radi pod karticom, svi podaci na disku se pri korišćenju defragmentatora uništavaju i nekom Norton Disk Doctoru nema ni govora! U Infochip System-u već pišu sopstvena oruđa koja će delovati na kartici Expanz!

Bez štetu se mogu aktivirati programi za oživljavanje podataka, ali oživljavanje datoteke neće ići kao što želite. Računar otkazuje poslušnost, kada zajedno sa Expanz!-om aktivirate DOS-ov CHKDSK. Zato uz karticu daju programčić koji stari CHKDSK zamenjuje novim. Takođe se zamenjuje stan DIR koji za svaku datoteku pokazuje postotak kompresije.

Expanz! nudi, za malo para, veliko proširenje diska, ali su sporedni efekti verovatno isuviše veliki da bi karticu sa radošću sebi priuštiti.

Za radnjače još adresa: InfoChip Systems, 2840 San Tomas Expwy., Santa Clara, CA 95051.



Nekada su diskovi imali vreme pristupa 28 ms...

A tada je ovanap dan kada se pojavio prvi disk sa vremenom pristupa od 8 ms. Razvila ga je firma Plus Development Corp. Čeo disk je smeštan na jednu dugačku 16-bitnu karticu, Hardcard II XL, svojim vremenom pristupa od 8 ms i prenosom vanrednih 1,4 Mb u sekundi, čak

osam do devet puta brži od standardnih diskova sa 28 ms. Postoje dva tipa - sa 105 Mb formatiranog kapaciteta. Očito nije daleko vreme kada će memorijski medijumi imati skoro jednako vreme pristupa kao i serijski memorijski čipovi. Ovak, radeci tako brzim diskom, uopšte ne primećuju bilo kakav zastoj pri upisivanju podataka. Instalacija Hardcard II XL je trenutna - kartica se stavlja u prazan utičak matičnoj ploči, prebacuje se na podešavanje CMOS tip diska i odmah se može početi sa radom. Za više informacija pozovite 800-624-5645 u SAD.

Izuzetni kapaciteti digitalnih kaseti

Konačno su ne tržišta došli nosači podataka koji za memorisanje koriste digitalne kasete DAT. Ove kasete su pre tri godine razvili Japanci kao čitajuć/upisujuć medijum koji direktno presnimava digitalni zapis sa laserskih diskova. DAT se u sudu tehnici (očevidno) ne čini brzo, ali sigurno je, da je vanredno prikladan za računarstvo. Digitalni zapis na traci, brz prilaz i veliku postojanost, želi svaki sistemski inženjer od medijuma za rezervne kopije (back-up). Možda je DAT baš to - nudi prenos 11 Mb u minuti, a u burst načinu čak do pet Mb u sekundi! Ovakve brzine su za zaštitu podataka kao porudžina, pa je zato Maynard Electronics razvio MaynStream 1300 DAT koji radi i pod Novellovom mrežom NetWare. DAT u jednoj kaseti smešta 1300 Mb, a kasete košta približno 70 USD. Ako se izuzme da se trako (među njima i DAT) strašno dugo formatiraju, onda je to idealan zaštitni medijum za svako bolje preduzeće.



Pažnja, visoki napon!

Kad veza između računara i štampača jednostavno nije moguća bez razbijanja zida ili više desetina metara ometajućeg kabla, možda će se neko, u očajanju, odlučiti za rešenje koje nudi GEC-Marconi Software Systems. Na izgled jednostavno rešenje, ali u stvari zatrebašujuće: štampač i računar se povezuju direktno preko električne mreže! Predajnik šalje u električnu mrežu signale koje prijemnik na drugom spratu predaje štampaču. Ovakav način prenosa je moguć do 10 metara, a omogućava »bežično« priključenje do sedam računara na jedan štampač. Svaki računar preko električne mreže predaje svoj kod, a štampačev prijemnik precizno zna ko je ko. Kažu da stvar ne utiče na ostale aparature koje su priključene na električnu mrežu, jer se komunikacija odvija na drugim frekvencijama električne struje. Nažalost proizvođači ne kažu šta se dešava kada električna struja u mreži veniše odnosno u nju udara grom. Tu verovatno ni UPS ne pomaže. Ako se odlučite za rad sa VERRAN AC Datafink/DPS, u svaki slučaj stavite rukavice i obujte cipale sa gumenim đonovima. Čovek nikada ne zna...

Za nabrajanje još kontaktna adresa: GEC-Marconi Software Systems, Suite 450, 12110 Sunset Hills Road, Reston, Virginia 20190.

Standard budućnosti?

MIRKO MAHER

Uvod

Programi sa prozorima su, do pre par godina, bili retkost i veliki luksuz, a danas je ovo pomoćno sredstvo uslov za uspeh programa na tržištu. Današnji sistemi prozora za personalne računare nisu kompatibilni, pa je zato prenos programa iz jednog na drugi operativni sistem mukotrpan i skup zadatak čak i za najveće proizvođače softvera. Većina jednostavno ne može sebi priuštiti da obezbedi podršku različitim sistemima (čak se i Borland ograničio samo na DOS).

Ako se prosečna aplikacija na PC-u temelji na prozorima, onda bi dobro bilo da se prozori standardizuju. Trenutno postoje bar tri različita sistema prozora – GEM i Microsoft Windows u DOS-u i Presentation Manager u OS/2. Idealno bi bilo kad bi standardni sistem prozora funkcionisao u raznim operativnim sistemima i na različito velikim računarima. Onda bi se mogle aplikacije jednostavno prenositi i na potpuno drugačiji hardver (prevenatno tip procesora). Pošto se proizvođači navedenih sistema prozora grozničavo bore za što veći deo tržišta, prirodno je da se od njih ne može ni očekivati racionalno rešenje – čitaj standardizacija. Ono se možda već javlja na drugoj strani. Zove se X Window System i razvijen je na MIT-u (Massachusetts Institute of Technology) za Unix.

Malo istorije

Kada je Unix sredinom sedamdesetih godina postao opšte poznat, najčešći korisnički interfejs bio je tekstualni videoterminal. Komunikacija sa računarem se odvijala tako da su se naredbe i odgovori računara pojavljivali u dnu ekrana, pomicali se nagore i nestajali na vrhu; to je bilo malo poboljšanje u poređenju sa terminalima koji su ispisivali na papirne rolne.

Kasnije su razvijeni savremeniji videoterminali, koji su se pokoravali mnoštvu naredbi gde će se i kako tekst ispisivati na ekranu. Nažalost, razni proizvođači koriste različite skupove naredbi (naravno da se nešto drugo ne može ni očekivati), ali samo Unix je toliko lukav da zna sve te razlike da usaglasi. Zato nije po-



trebno da programar poznaje sve moguće i nemoguće tipove videoterminala. Umesto da se tekst ispisuje direktno na ekran, kontroliše se sistematskom programskom bibliotekom.

Unix je multitasking operativni sistem, što znači da se u njemu može istovremeno obrađivati više programa, ali njihovo organizovano ispisivanje na ekranu nije bilo standardizovano. Gornji sistemi prozora koriste sposobnosti inteligentnih videoterminala za odvajanje na ekranu izlaza različitih programa. Od ostalih poboljšanja, razvijeni su grafički videoterminali. Istorija se, naravno, ponavlja. Svaki proizvođač ima svoj skup naredbi, čak i među pojedinim modelima istog proizvođača postoje male razlike koje takođe zahteva-

ju male izmene u programskom kodu za upravljanje ispisu na ekranu. Sa svim tim su se u koštac uhvatili istraživači na MIT-u, koji su počeli sa realizacijom projekta Athena.

Za rešenje tog problema, osnovali su projekat X. Rezultat je odlično koncipirana okolina ekranskih prozora. Napredovanjem projekta X, povećavali su se i njegovi ciljevi. Nastao je proizvod koji daleko premašuje kapacitete konkurenata u ličnim računarima. Trenutna verzija ne samo da podržava skup različitih terminala, već omogućava i distribuirano izvođenje aplikacija. To znači da nije neophodno da se program odvija na računaru na koji je terminal stvarno priključen, već bilo gde u računarskoj mreži.

X može da prikaže izlaze više pro-

grama koji istovremeno rade na jednom ili više računara. Korisniku omogućava rad različitim računarima, kao da je terminal istovremeno na njih priključen. Pri tome nije neophodno potrebno da su računari jednaki, niti da imaju jednake operativne sisteme.

Kako radi X

Kod tradicionalnih grafičkih sistema, aplikacija koja hoće da nešto nacrtaj na ekran, poziva odgovarajući potprogram koji je po pravilu deo sistema ili grafičke biblioteke. Potprogram izvodi zahtevani zadatak (npr. nacrtaj liniju) i vraća kontrolu aplikaciji. (Slika 1.)

Sistem X radi tako da kontroliše ekrane ustupa posebnoj aplikaciji koja se zove SERVER X (u daljem tekstu server). Aplikacija koja želi nešto da nacrtaj, poručuje serveru, slanjem informacione «poruke» sa opisom zahtevanog zadatka. Po slanju poruke, kontrola se vraća aplikaciji. Odgovor servera nije obavezan. Skup važućih poruka zove se PROTOKOL X. Takve poruke su na primer: nacrtaj liniju, nacrtaj krug ili ispiši tekst. Aplikacija koja formira svoj grafički izlaz porukama protokola X, zove se KLIJENT X, u daljem tekstu klijent. (Slika 2.)

Poruke ne idu samo od klijenta prema serveru, nego i u suprotnom pravcu. To su poruke o događajima (npr. pritiscima na tastaturu i škljocanju miša) ili o greškama. Ove poruke su, takođe, deo protokola X. Prozora na ekranu, deo protokola X. Prozora na ekranu, deo protokola X. Prozora na ekranu, deo protokola X. Naravno, i server može usluživati više klijenata odjednom. Njegov zadatak je samo da razume poruke protokola X.

X je sistem koji gone događaji (event-driven). To znači da klijent čeka dok se na serveru ne pojavi akcija koja ih interesuje. Tada server pošalje klijentu poruku – npr. ponovo nacrtaj prozor, dimenzije prozora su izmenjene ili pritisnuta taster na tastaturi.

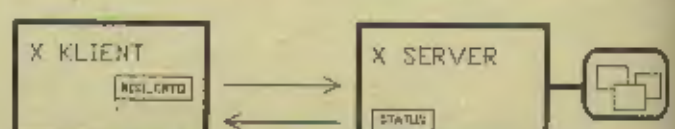
Klijent reaguje tako da napravi potrebne izlazne poruke i vrati se u stanje čekanja na sledeći događaj.

Ova «filozofija» je drugačija od one kod tradicionalnih aplikacija. One su orijentisane proceduralno u komunikaciji sa korisnikom preuzimaju aktivnu ulogu i vode ga kroz program. Ulazne podatke (pritiske na tastaturu ili škljocanje miša) primaju samo u odrađenim trenucima. Tipična proceduralna aplikacija je na primer unos u bazu podataka.

Aplikacija događaja je pasivna. Svakog momenta je spremna da od-



slika 1 - tradicionalni grafični izlaz



slika 2 - X Window grafični izlaz

govori na zahtev korisnika ili sistema. To je fleksibilan okvir u kojem radi korisnik. Tok rada nije unapred određen, do cilja se može doći različitim putevima. Korisnik bira između ponuđenih oruđa dok ne postigne željeni rezultat. Programi za crtanje (npr. Paintbrush ili AutoCAD) tipični su predstavnici takvih aplikacija.

X kao distribuirani sistem

Pošto server sa klijentima komunicira porukama, jednostavno se mogu slati, preko računarske mreže, zahtevi protokola X i serveru koji radi na drugom računaru. Sistem X Window je zasnovan na porukama baš zato da bi bio mrežni grafički sistem. (Slika 3.)

Slika 3 prikazuje da klijent koji se izvodi na računaru A, koristi (preko servera na računaru C) ekran računara C. Klijent koji se izvodi na C, koristi ekran računara B (preko servera na B) gde radi i klijent koji koristi ekran vlastitog računara. Sistem X Window je idealna osnova za razvoj svamog distribuiranih aplikacija.

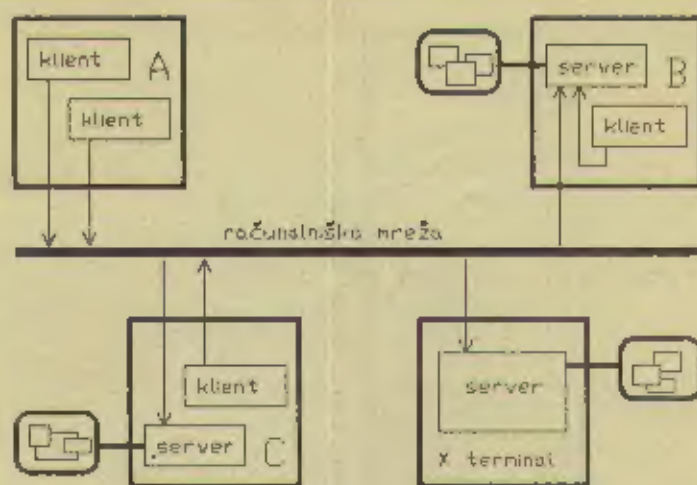
Ali, sa tim dobrotama još nije kraj. Upotrebjeni računari u mreži mogu biti od raznih proizvođača, sa različitim procesorima i da na njima rade različiti operativni sistemi. Komunikacija se u mreži odvija pomoću poruka po precizno definisanom protokolu X. Program koji recimo radi u Unixu, ne može se prekopirati na računaru sa DOS-om i tamo se izvoditi (treba preneti izvorni kod i program ponovo prevesti u drugi operativni sistem). X Window je, zadata, jedini otvoreni distribuirani grafički sistem.

Na slici 3 imamo primer kad jedan od klijenata računara A koristi ekran posebnog računara na kojem radi samo server u funkciji udaljenog terminala. Ovakav tip računara zove se terminal X i njegov jedini zadatak je prikazivanje izlaza klijenata koji rade na drugim računarima. Većina aplikacija sistema X Window na PC-ima, zbog ograničenja DOS-a realizovane je samo kao terminali X, jer već server zauzima celu memoriju. Osim toga, DOS je sistem koji obrađuje samo jedan zadatak istovremeno: ako treba na istom računaru da rade server i bar još jedan klijent, potreban je multitasking sistem. Više o tome kasnije.

Server zna, prema zahtevima protokola X, da uradi samo grafički izlaz, a korisniku ne daje oruđa za određivanje veličine, položaja ili prekrivanja prikazanih prozora. To se može uraditi u svakom od klijenata, ali se svaki put moraju ponovo programirati. Mogu se realizovati i na serveru, ali su se autori odlučili za još fleksibilniji zahvat.

Upravljač prozora

Svakom serveru pripada poseban klijent, window manager, na našem jeziku upravljač prozora (koji radi na lokalnom ili udaljenom računaru). Ovaj program ima posebne privilegije i sme da kontroliše prikazane prozore. Upravljač prozora zauzima prikazane prozore ostalih



slika 3 - distribuirani sistem



klijenata nekakvom dekoracijom koja sadrži ime prozora i dugmeta za pomicanje i menjanje veličine prozora. Korisnik škljocanjem miša poručuje gde želi da ima prozor, a zadatak upravljača prozora je da to ostvari.

Trenutno postoji više upravljača prozora. Najpoznatiji su verovatno OSF/Motif i OPEN LOOK. Upravljači prozora zasnovani su tako da se u radu jedan može zaustaviti a drugi aktivirati. Unutrašnjost prozora se ne menja, menja se samo dekoracija oko njih.

Ako klijent želi da komunicira sa serverom, mora da generiše odgovarajuće poruke. To može biti dangubno, pa su zato izradili programsku biblioteku Xlib. To je najniži interfejs koji klijent može da koristi. U stvari, to je mnoštvo potprograma u C jeziku, koji većinom šalju serveru po jednu poruku. Takav potprogram npr. nacrtati jednu liniju. Ali Xlib ne određuje oblik aplikacija odnosno njihovih prozora na ekranu. Iako se klijent može programirati Xlibom, programeri u većini slučajeva koriste dodatnu biblioteku programskih oruđa (toolkit), koja određuju i oblik.

Takvo programsko oruđe je interfejs na višem nivou koji zna da generiše npr. menije, dugmeta (padding), ključače i slično. Obično se jedan poziv potprograma takve biblioteke preslika u nekoliko poziva Xliba i, samim tim, u nekoliko poruka serveru. Navodimo neke najpoznatije zbirke oruđa:

Athena Toolkit - prilično osnovna zbirka oruđa, napravljena na MIT-u.

OSF/Motif Toolkit - zbirka oruđa koje aplikacijama daje prividnu trodimenzionalnost. Promoviše je konzorcijum preduzeća udruženih u OSF (Open Software Foundation) u kojem, između ostalih, saraduju DEC, HP i Microsoft.

Xol Toolkit - temelji se na standardu OPEN LOOK. Isporučuje ga AT&T, zajedno sa odgovarajućim upravljačem prozora.

Xview Toolkit - takođe se temelji na standardu OPEN LOOK, ali koristi drugi programski interfejs nego Xol. Proizvođač je Sun Microsystems. (Napomena: OPEN LOOK nije zbirka oruđa ili upravljač prozora, nego samo precizna specifikacija korisničkog interfejsa; njoj su određeni Xol i Xview i zato imaju isti oblik.)

XVT Toolkit - ovaj je posebno zanimljiv, jer programi, napravljeni njime, koriste biblioteku Xlib i OSF/Motif ili OPEN LOOK i zato mogu biti različitog oblika. Mogu se koristiti i biblioteka za macintosh, Microsoft Windows ili Presentation Manager (OS/2). Tako se mogu napraviti aplikacije koje rade i u drugim okolinama. (Slika 4.)

Sistem X Window nudi niz zanimljivih i prvenstveno korisnih mogućnosti, ali se prosečnom jugoslovenskom korisniku, kojem se pogled (čitaj dubina džepa) proteže samo do PC-kompatibilaca, sve to više čini kao naučna fantastika DOS za-

pravo postavlja hrpu ograničenja koja bar na prvi pogled, na samom početku, obeshrabruju da bi se u njega mogao smestiti sistem kao što je X Window. Ali izgled lakše i da vara. Kako bismo prevaru lakše otkrili, prvo ćemo detaljno pregledati navedena ograničenja.

A šta DOS?

Svi koji imate DOS u malom prostoru, preskočite sledećih nekoliko pasusa! Ostali ćemo osvežiti pamćenje i pogledati kako je u DOS-u podeljena memorija (RAM). Konvencionalna memorija je od 1 do 1024 K (1 Mb). Pri dnu su DOS, pogoni uređaja (device drivers) i prijavni programi (TSR). Grafički RAM zauzima područje od 640 do 768 K, iznad vrha DOS-a i ispod grafičkog RAM-a nalazi se aplikacijsko područje. Iznad 768 K je sistemsko područje gde se nalaze sistemski BIOS i neke slot kartice (npr. za mrežu). U sistemskom prostoru nalazi se uvek nekoliko neupotrebljenog adresnog prostora, nekada su to čak komadi po 128 K (Slika 5.)

Produženi (extended) RAM je adresni prostor iznad 1 Mb i seže kod procesora 286 do 16 Mb, a kod 386/486 do 4 Gb. U osnovnom procesoru 8086 ne postoji. Program koji želi da koristi taj deo memorije mora procesor da prebaci u takozvani zaštićeni režim rada (protected mode), koji nije kompatibilan sa DOS-om i normalnim aplikacijama koje se odvijaju u realnom režimu (real mode).

Prošireni (expanded) RAM radi kao rezerva koja se pod kontrolom posebnog programa (expanded memory manager) i hardvera preslikava u konvencionalnu memoriju. Ovde se ne radi o prepisivanju delova memorije, već o pomicanju kazaljki koje omogućavaju da se deo ove rezerve prividno prikaže, na određenoj adresi, ostalim programima. Suprotno produženoj memoriji, proširena je na raspolaganju kod svih tipova procesora. Poznajemo tri vrste proširenih memorija: EMS 3.2 koja preslikava u sistemsko područje memorije najviše 4 stranice po 16 K, te EMS 4.0 i EEMS koje preslikavaju više proizvoljno velikih područja bilo gde u konvencionalnu memoriju.

Procesori 386/486 sa svojim dodatnim osobinama takođe omogućavaju programsku emulaciju EMS memorije. Jedan od programa koji to omogućava jest QEMM-386 koji produženu memoriju pretvara u proširenu. To je i najjeftinija varijanta, jer nije potreban hardver za preslikavanje memorijskih stranica (zapravo, već se nalazi u procesoru).

Kao što smo već rekli, DOS je operativni sistem koji obrađuje samo jedan zadatak (task) istovremeno. Špretnim programiranjem uspešno je da se izbegne to ograničenje, pa niz programa omogućava multitasking. Na primer, SlideKick može privremeno da prekine izvođenje trenutnog programa. Takođe postoje programi (npr. DESQview) koji omogućavaju pravi multitasking. Na primer: dok u Wordu otkucavate

tekst, Quattro u pozadini proračunava tabele.

Iako DOS ograničava memorijski prostor za aplikacije na 640 K, istovremeno se mogu proširenom memorijom aktivirati programi koji svi zajedno zauzimaju mnogo više memorije. Kad se programi preklapaju, preklapaju se i stranice EMS memorije (samo u EMS 4.0 ili u programskoj emulaciji na 386/486, gde broj strana nije ograničen). Naravno da za pojedini program mora ostati dovoljno memorijskog prostora, kad se od 640 K oduzmu DOS, pogoni uređaja i pritaženi programi. Retko je taj ostatak malo veći od 500 K. Pogoni i pritaženi programi moraju biti stalno prisutni i ne mogu se preklapati EMS-om. Nije neophodno da su baš smešteni u aplikacionom području memorije. Mogli bi biti i u sistemskom gde su, kao što smo već rekli, dosta velike «rupe». To su spretni programeri i uradili neki programi (npr. QRAM za procesor 286 i QEMM za 386) znajući da te rupe popune memorijom i u nju smeste pogone, pritažene programe i neka delove DOS-a (npr. BUFFERS i FILES koji su definisani u CONFIG.SYS i zauzimaju dosta memorije).

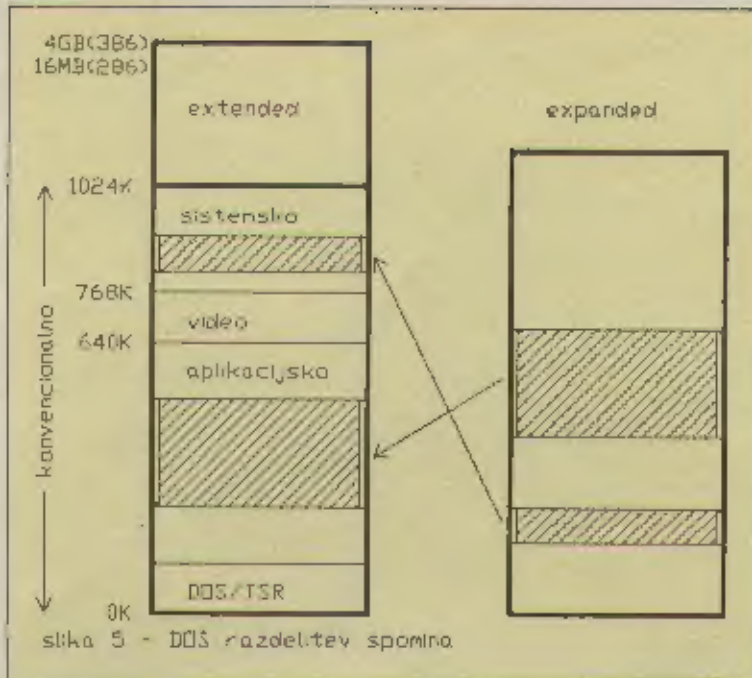
O lepom ponašanju

Do sada smo, stalno pričajući o multitaskingu, previdali (razume se, namerno) problem pisanja na ekran. Dosta retki su programi koji samo nešto računaju bez istovremenog izveštavanja šta zapravo rade. Neki ispisuju samo na jedan deo ekrana (u prozor), ali većina to radi na celom ekranu. Žato se istovremeno ispisivanje dva ili više programa na ekran mora uskladiti. To se može uraditi tako da se svima osim jednom programu zabrani pisanje na ekran ili da se svakom odredi izlazni prozor.

I sada smo u domenu ponašanja? Program se ispisuje na ekran na dva načina: pomoću DOS i BIOS potprograma (to ćemo okvalifikovati kao lepo ponašanje) ili direktno u onaj deo memorije gde je smeštena grafička kartica (to jest u područje između 640 i 768 K, zavisno od tipa kartice). To je loše ponašanje, koje se u neku ruku može i opravdati jer su DOS i BIOS potprogrami veoma spori. Žato većina proizvođača (ili programera) kod mnogih tekstualnih i skoro svih grafičkih aplikacija upražnjava drugi metod (možda ste kod Turbo Pascala ili Turbo C-a primetili promenljivu koja, u vašim programima, omogućava izbor između navedenih načina).

Multitasking programi (npr. DESQview) veoma lako ukroćuju aplikacije koje se lepo ponašaju. Oni preusmeravanjem prekidnih vektora zamenjuju sistemske potprograme za ispisivanje svojim potprogramima i ispis ograničavaju, prilagođavaju ili čak zabranjuju.

Veći problem su aplikacije koje se loše ponašaju zbog direktne upotrebe grafičke memorije. U procesorima 386/486 mogu se i takvi obešnjačiti ukrotiti. Ovi procesori u potpunosti podržavaju prividnu (virtualnu) memoriju. DESQview zna da iskoristi tako da deo neupotrebijena



slika 5 - DOS razdelitev spomina.

memorije podmetne aplikaciji kao grafičku memoriju, a zatim odgovarajuće «obrazane» podatke sam prekopira u manji prozor u pravoj grafičkoj memoriji. To važi kako za tekstualne tako i za grafičke aplikacije koje uopšte ne znaju da su uokvirene. Izuzetak su aplikacije koje procesor preklapaju u zaštićeni režim rada, ali o tome kasnije.

Problem je kod ostalih procesora (8088/86 i 286) rešiv samo ako minimalno saraduju i aplikacije. Pri startu treba, takozvanom naredbom SHADOW, proveriti da li stvar radi pod čistim DOS-om ili pod multitasking programom. Detalje ćemo izostaviti. Tako rade na primer Paradox, dBase i Word Perfect.

Od 1981. godine kada se pojavio prvi PC, za njega su korišteni različiti tipovi Intelovih procesora. Svi su oni imali sve funkcije prethodnog i na taj način su obezbeđivali kompatibilnost nadole, i takođe su nudili nove naredbe i mogućnost rada većom memorijom.

Prvi PC je imao procesor 8088. Iznutra, to je 16-bitni procesor koji može da po segmentima adresira 1 Mb memorije, a spolja se ponaša kao 8-bitni i radi u realnom režimu (real mode). Njegov naslednik (8086) razlikovao se samo po tome što je i spolja bio 16-bitni procesor.

Procesor 286 (to je bio sledeći model) nudi nov, zaštićeni režim rada (protected mode). Adresuje (opet po segmentima) 16 Mb memorije. Pored višestrukog rada, zaštićeni režim rada nudi međusobnu zaštitu programa (po tome je i dobio ime). Ako se «raspadne» jedan program, to ne utiče na rad ostalih programa i operativnog sistema. Nažalost, zaštićeni režim rada je sasvim nešto drugo od realnog (u kojem se odvija DOS), pa ga dugo nije koristila nijedna aplikacija, tako da je upotreba procesora 286 značila samo brži rad. Položaj se promenio tek kad su nastali programi zvan «proširivači DOS-a» (DOS extenders), ali o njima kasnije.

Za njim došao procesor 386. Pored svih svojstava svojih prethodnika, doneo je nov režim rada. To je prividni režim 8086 (virtual 8086 mode) koji je u stvari podrežim zaštićenog. U zaštićenom režimu rada nudi 1 Mb kao i 8086. Ovo elegantno re-

šenje omogućava istovremeni rad više nepromenjenih realnih (DOS) programa. Inače, to je 32-bitni procesor koji adresuje 4 Gb memorije (segmentno ili direktno). Navedimo još 386SX koji je programski ravnopravan, ali je spolja 16-bitni.

Poslednji procesor je 486. To je u stvari samo brži procesor od 386; u istom čipu ima još matematički koprocessor i prihvatnu memoriju (cache).

Proširivači DOS-a

Kao što smo već rekli, DOS radi samo u realnom režimu i zato ne može da koristi više od 1 Mb memorije, iako to procesor dozvoljava. Proširivač je nekakva nadgradnja aplikacije koja hoće da radi u zaštićenom režimu, a dodaje se pri povezivanju programa.

Ideja je sledeća: kad program treba da pozove potprogram u DOS-u ili kad ima bilo kakav drugi zahtev koji je ostvariv samo u realnom režimu, tada proširivač prekopira potrebne podatke u područje ispod 1 Mb, prebacuje procesor u realni režim i poziva zahtevanu funkciju, a zatim procesor ponovo prebacuje u zaštićeni režim i programu vraća rezultate.

Razlikovati treba proširivač koji se odvija u procesoru 286 (sa pristupom do 16 Mb) od onog kojeg zahteva 386 (sa pristupom do 4 Gb). Obično se nudi i opcija prividne memorije, što znači da program radi, iako nema na raspolaganju dovoljno memorije. Proširene DOS aplikacije imaju dva dela. Realni deo proširivača nalazi se u konvencionalnom delu memorije, a zaštićeni deo proširivača i aplikacija u proširenom. Prvi deo je dosta manji od drugog, a ograničenje memorije (640 K, ako se još sećate) ne smeta istovremenom radu velikih aplikacija, jer je dovoljno područje memorije koje pokriva zbir realnih delova korisničkih aplikacija.

Ali kad se procesor preklopi u zaštićeni režim, proširivač postaje nekakav sistemski upravljački program. Više upravljačkih programa ne može istovremeno da radi bez konflikata. Žato su se firme Quarterdeck i Phar Lap (koje su takođe proizvođači najpoznatijih proširiva-

ča) dogovorile za specifikaciju VCP (Virtual Control Program Interfaci) za procesore 386; nju su kasnije uvojili i ostali. Jednostavnije rečeno, upravljanje preuzima program koji je prvi startovao, a ostali mu prepuštaju upravljanje kada im je potreban realni režim. Svaki program mora pri startu da proveriti da je neko bio pre njega, pa tek onda preuzima upravljanje.

Pomenimo i sličan standard za 286, to je XMS, ali se u njegovim detaljima nećemo upuštati.

Sadašnjost...

Još na početku smo naveli neke sisteme prozora koji su sada u upotrebi. Pogledajmo neka njihova svojstva, prvenstveno mogućnost multitaskinga!

■ GEM-u rade samo oni programi koji su pisani za njega. To je verovatno i razlog zašto sistem nije suviše raširen. Nepoznat mu je pravi multitasking. Može se prekinuti rad u jednom programu, izvesno višim raditi u drugom programu, vratiti se u prvi, te produžiti rad tamo gde je bio prekinut, ali je prekinuti program za to vreme čekao.

Windows nudi pravi multitasking kako za programe koji su za njega pisani, tako i za obične DOS programe. Ali DOS programi uvek zauzimaju ceo ekran ili na njega uopšte ne ispisuju. Prozori su lepi jer ih je zamislio stručnjak za dizajn.

DESQview pored prvog multitaskinga prikazuje, u prozorima, iza standardnih DOS programa. U njemu istovremeno radi više proširenih programa (u zaštićenom režimu). Sve zavisi od procesora, lepo ponašanja i vrste programa (tekst/grafički). ■ procesoru 8086, u prozorima rade samo tekstualni programi koji se lepo ponašaju, dok ostali rade istovremeno, ali na celom ekranu. Isto je i sa 286, samo da su uključeni i prošireni programi (naravno za 286). U 386 se mogu, u prozorima, odvijati istovremeno svi realni programi (i grafički, bez obzira na ponašanje) i neki prošireni (oni koji uvažavaju VCP), a ostali prošireni samo na celom ekranu. Spomenimo još da oblik prozora mnogo zaostaje za potpunim iskoriscenjem svih mogućnosti procesora.

I na kraju još OS/2 Presentation Manager. Pošto je OS/2 pisan za zaštićeni režim (zato i zahtev za bar 286), u njemu je multitasking takođe ugrađen. OS/2 poznaje takozvani način udruživanja prozora sa DOS-om u kojem rade i standardni DOS programi, ali su poteškoće pri udruživanju dosta česte. Oblik prozora veoma potseća na Windows, što nije iznenađujuće, jer je obadvojica izradila Microsoft. Iznenađujuće je to da OS/2 niti u SAD ne može da osvoji značajniji deo tržišta (razlog je bar u početku bio nedostatak programa za njega), jer u suštini nudi više od DOS-a. Istina je i to da zahteva veću memoriju (preporučivo je bar 4 Mb) i zauzima više diska. Možda je uzrok i to da prosečan korisnik još ne zna DOS pošteno da koristi.

... i budućnost

Spomenuti Quarterdeck je već nekoliko puta najavljivao, za prvi kvartal 1991. godine, DESQview/X za koji tvrde da će "DOS produžiti u 21. vek". Navodno da testna verzija već teče bez problema. Najavljuju mnogo više od implementacije servera X. Kažu: »Ponudićemo razvojni sistem ■ aplikacije (Klijente) X Window u DOS-u i mogućnost izvršavanja klijenata X u DOS-u ili računarskoj mreži. Budući da najavljuju pravu revoluciju u DOS-u, pogledajmo detaljnije njihova obećanja!

DESQview obezbeđuje multitasking, pod kojim se odijela i server koji podržava ekrane EGA, VGA, prošireni VGA, 8514 i DGIS. Server radi kao realna ili proširena DOS aplikacija, pri čemu druga varijanta obezbeđuje više prostora i samim tim više istovremenih prozora na ekranu. Standardne DOS aplikacije (WordPerfect ili Lotus 1-2-3) pod DVX (DESQview/X Translation Software) se ponašaju kao normalni klijenti X. U stvari, DVX presreće njihov izlaz na ekran i preformira ga u naredbe protokola X. To navodno radi tako brzo da korisnik ne primećuje nikavo zakašnjenje. Potrebno vreme zavisi i od vrste aplikacije (lepo ponašanje, tekstualna ■ grafička, realna ■ proširena aplikacija) i tipa procesora. Različite kombinacije su jednake kao i kod već poznatog programa DESQview, samo da po novom izlaz ide preko servera X.

Bez poteškoća se izvode programi koji su za DOS realizovani kao klijenti X: veličina programa koji se izvodi, ograničena je samo tipom procesora (16 Mb ■ 286 i 4 GB u 386). U stvari, ovih programa još nema (verovatno Quarterdeck ima neki primerak za testiranje), ali verovatno na njih nećemo dugo čekati, jer će ih jednostavno preneti iz Unixa.

Kao što smo već rekli, kontrolu nad prozorima vrši poseban klijent, upravljač prozora. Sastavni deo paketa DESQview/X je i vlastiti upravljač prozora, zvan DWM, koji se odlikuje po tome da zauzima samo 50 K memorije. U poređenju sa njima, ostali (npr. OSF/Motif ■ Xol OPEN LOOK), koji inače dobro rade, zauzimaju mnogo više memorije (pre navedeni: 600 K i 1,5 Mb), a to ■ važno.

Najzanimljivije mogućnosti DESQview/X obećava PC-ima koji su povezani ■ mrežu. DESQview podržava i najrašireniji sistem PC mreže, to je Novell-Netware sa protokolom IPX. Pri startu klijenta X, jedan od parametara pokazuje na kojem će se ekranu radnja odigravati (to je standardan postupak sistema X). Ako to nije lokalni ekran, DESQview/X preko mreže usmerava naredbe protokola na odgovarajući račun. Ako je to lokalni ekran, naredbe šalje lokalnom serveru, dok naredbe koje dolaze preko mreže sa udaljenog računara, usmerava na lokalni server, a ovaj ih prikazuje na ekranu.

Korisnik može da startuje lokalno i udaljene klijente. Pošto ■ mreži lako sarađuju DOS i Unix računari (kao i ostali), otvaraju se mnoge zanimljive kombinacije. Na primer:

ako se Unix mreži doda jak računarski DOS (386 sa dovoljno memorije), onda DVX omogućava korisnicima rad sa mnoštvom standardnih DOS programa. Važi i obrnuto: jedan Unix računarski u DOS mreži znači da svi korisnici mogu da rade kako sa DOS tako i sa Unix programima. I treća varijanta: kad se mreži PC-a 286 (teoretski čak 8086, tako ■ ovi računari u praksi za tu namenu presport) doda jedan jak 386, tada svi korisnici imaju mogućnost da izvršavaju velike proširene programe koji inače prevazilaze kapacitet računara 286. Korisniku je na kraju krajeva sasvim svejedno na kojem računaru se njegov program izvršava ■ lokalnom ili udaljenom. DESQview/X zna taj detalj da sakrije. (Slika 6.)

Slika 6 prikazuje primer sistema DESQview/X sa DWM («remote» i «local» je napisano samo zbog jasnoće, obična aplikacija nema te oznake). Klijenti DOS 128 K (COMMAND.COM) i Lotus 1-2-3 odvijaju se na udaljenom računaru DESQview/X. Borland Turbo C++ i Application Manager (označen sa main) lokalni su klijenti. Xlock i Xterm su udaljeni klijenti koji se odvijaju u sistemu SCO Unix. Rada još četiri klijenta (lokalno i udaljeno) koji su na ekranu označeni samo simbolima na desnoj ivici.

Quarterdeck preporučuje za DESQview/X kao minimum 286 i memoriju od 2 Mb (teoretski radi čak u manjim sistemima). Još bolje je (zbog boljeg rada sa memorijom) ako se koristi računarski sa procesorima 386/486. Koliko memorije je potrebno, izračunava se tako da se sabere količina potrebna za pojedine programe, koji će istovremeno raditi, i doda 1 Mb za sistemske programe.

Na kraju, pokušajmo da odgovorimo na pitanje postavljeno u naslovu. X Window System ima pred konkurencijom bar jednu veliku prednost: jedini je koji nije vezan na operativni sistem, a njegova distribucija ne zavisi od komercijalnih interesa jednog proizvođača. Ko god hoće da ga instalira na bilo kakvom sistemu, može besplatno dobiti izvorni kod odnosno kupiti ga po ceni kopiranja na MIT-u. X Window System može da radi u svakom sistemu koji podržava multitasking i komunikaciju između programa i ima interfejs za C. To je kod Unixa već standard, jer pod AIX radi u PS/2, SCO Unixu, Interactive Systems 386/ix i većim sistemima, kao što su VAX, Cray i Data General serija MV.

Da li onda treba razvijati da pišu programe za čisti DOS ili za okollnu prozora? Verovatno je čisti DOS stvar prošlosti, dok je za prozore odluka teža. Pogrešan izbor bi, osim za najveća preduzeća, bio sudbonosan. X nudi dosta perspektivnu mogućnost, pa će se za njega odlučiti mnogi razvijaci. Bitne će biti odluke najvećih proizvođača. Ako ga bar neki usvoje, ima osiguranu budućnost, a ostali će se sistemi prozora naći u nezavisnosti. Čak i veliki mogu zapasti u teškoće ako ne podrže X, jer je razlika između personalnih računara i radnih stanica sve manja.



IDenticus Slovenija d.o.o.

Podjela za proizvodnju in zbranje računalskih, opreme za automatsku identifikaciju in storitve. Trgovanje preko štirideset mednarodnih i domačih referenci na področju avtomatske identifikacije. Nudimo rešenja po sistemu KLJUČ U RUKE.

U svojih rešenjihna nudimo opremo slededih proizvođača:

DATALOGIC, Italija (oprema za čitanje linijskog koda)

- industrijski laserski čitači
- prenosni računari PC32
- dekodirni linijskog koda

OPTICON, Japan (oprema za čitanje linijskog koda)

- svetlosna pera
- CCD čitači
- ručni laserski čitači sa VLD diodama

DH-PRINT, SAD (termalni štampači)

- DH-P 324 low cost termalni štampač
- THARO, SAD (štampač linijskog koda)
- termal transfer štampači grafike i linijskog koda
- continous laserski štampači grafike i linijskog koda
- EASYLABEL programska oprema za ispisivanje linijskog koda i grafike

CAERE, SAD (oprema za čitanje OCR znakova)

- OCR proreznici čitači
- magnetni čitači ISO trag 1 i trag 2
- OMNIPAGE, SW za prepoznavanje teksta
- DFI**, Tajvan (periferni uređaji)
- 400 dpi handy skeneri
- miševi

SPECTRA-PHYSICS, SAD (POS laserski čitači)

- model 750 SL
- model FREEDOM

LOGIKA COMP, Italija (embosirni i kodirani uređaji)

SPECIJALNE ETIKETE SA LINIJSKIM KODOM

proizvođača COMPUTYPE, SCHNOOR, METALCRAFT za:

- krevne banke
- biblioteke
- označavanje osnovnih sredstava
- identifikaciju brojeva vnde, gasa i struje
- elektronsku industriju
- tekstilnu industriju

Garancija za svu navedenu opremu po principu zamene sa ekvivalentnom opremom za vreme kvana. Tražimo posrednike. Mogućnost prodaje ■ OEM principu. Količinski i posrednički popusti. Drugo izdanje knjige **AUTOMATSKA IDENTIFIKACIJA ARTIKALA** (120 strana na slovenačkom jeziku). Cena knjige ■ 1.000,00 din

Firma IDenticus Slovenija d.o.o. je član Međunarodnog udruženja proizvođača opreme za automatsku identifikaciju AIM EUROPE.

IDenticus Slovenija d.o.o.
CELOVŠKA 108, 01107 LJUBLJANA
JUGOSLAVIJA
tel. +38 61 554-205, 557-655
fax. +38 61 57-407

OBJEKTIV

»OBJEKTIV« – prva jugoslovenska revija za fotografiju, video i hi-fi.

PIS BLED d.o.o.

podjovne prostorije: Kumerdoleva 18, BLED
FAX/TEL: (064) 78-170, ponedjeljak – petak, od 7.00 – 15.00

RAČUNARSKA OPREMA

- PIS 266/12 kupovina: 24.990 din, leasing 3.190 din
- PIS 386/16 NEAT kupovina: 26.990 din, leasing 3.590 din
- PIS 386/16 SX kupovina: 29.990 din, leasing 3.790 din
- PIS 386/25 CACHE 64 Kb kupovina: 44.990 din, leasing POZOVITE
- PIS 486/25 CACHE 128 Kb kupovina: POZOVITE, leasing POZOVITE
- štampač EPSON LQ 850 – kupovina: 24.998 din, leasing 3.190 din
- štampač EPSON LQ 1050 kupovina: 28.990 din, leasing 3.790 din
- leasing: 12 mesečnih rata bez depozita, poslednja rata otpisi
- Sva oprema sadrži: 1 MB RAM, brzi disk HD, 44 Mb (24 ms), kontroler WD udružuje sa
- 1,2 JAPAN – kućice sa uređajem za napajanje
- štampač NEC P7 – 19.990 din
- hard disk Fujitsu 160 Mb + kontroler ■ 1-1 22.990 din
- hard disk NEC D3142 44 Mb (24 ms) MFM 7.990 din
- VGA monitor 14" 1024x768 9.990 din

PROGRAMSKA OPREMA

- vođenje AUTOKAMPOVA
- linijalno poslovanje, komercijalna poslovanje
- za ostalo POZOVITE
- SERVIS za vođenje poslovnih knjiga za zavariva i privatnu proizvodnju
- održavanje, sklapanje, mreže, savetovanje
- Sva zrna sa besplatno na prijelaz, fax, BLED, u toku isporuke od 5–30 dana

Da li je rad računarom opasan?

PRIMOŽ GSPAN

Na pitanje da li je rad računarom štetan po zdravlje, odgovor nije jednostavan. Da li određena vrsta posla škodi ili ne odlučuje niz faktora: vrsta i način rada, uslovi u radnoj prostoriji, radno mesto, mašinska oprema, programska oprema, organizacija rada, osposobljenost radnika i njegovo zdravstveno stanje. Istina je da je operateri često žalili na razne tegobe, među kojima najčešće pominju sledeće:

- tegobe s očima (zamor, pečenje, slabljenje vida, smetnje u zapažanju boja itd.),
- glavobolje,
- bolove u skeletu,
- bolove u mišićima,
- nervozu,
- opšti zamor, loše opšte stanje i neraspoloženja.

Doduše, navedene tegobe nisu karakteristične samo za rad računarom, već su neprijatnosti te vrste poznate i u nizu drugih profesija. Od njih pate lektori, kontrolori boca u punionicama, krojačice u fabrikama konfekcije, daktilografkinje, radnici koji rade mnoge druge statičke poslove pri kojima naprežu oči i telo drže u prinudnom položaju. Takvi radnici svoj radni vek često završavaju odlaskom u invalidsku penziju. Upravo takvim poslovima do sada je neopravdano posvećivano manje pažnje nego što se danas posvećuje radu za ekranima računara.

U pravilu se greši već pri osposobljavanju za rad računarom, jer se na tim kursevima kandidati ne upućuju u moguće neprijatnosti i načine kako ih izbeći pravilnim izborom opreme, uređenjem radnog mesta i organizacije rada. Zato nije nikakvo čudo što čovek obično nailazi na tegobe tek tokom rada, što ga zatiče nespremnog. Uzrok tegoba ubrzo počinje da pripisuje raznim »zračenjima«, hvata ga panika i počinje da traži »zaštitne filtere«. Paniku uspešno pojačavaju proizvođači i prodavci filtera svojim reklamama, pa i razni alarmantni članci u laičkoj literaturi, koji prenose podatke iz neproverenih izvora ili na svoj način tumače inače korektna rezultata istraživanja.

Prvo pravilo, dakle, glasi: svaki početnički kurs računarstva treba da sadrži i uvodna uputstva za pravilan i zdrav rad računarom. Treba priznati da je rad računarom sa stanovišta zaštite na radu složen, jer ujedinjuje čitav niz nepovoljnih uticaja: to je izrazito statički rad u stalnom, često neprilodnom položaju tela; opterećuje vid, pri čemu je ugao gledanja ograničen, a rastojanje konstantno; radni ritam najčešće je nametnut; rad iziskuje palhički napor i koncentraciju, često je monoton, a s druge strane iziskuje često prilagođavanje novim načinima rada, odnosno novim programima; dijalog s računarom je bezličan, radnik ima malo kontakta sa sarad-

nicima i okolinom, a uz to često radi u vremenskom tesnacu. Svaki od tih uticaja posebno može da bude relativno bezazlen, ali njihova je kombinacija krajnje neprijatna.

Međutim, dejstvo svih tih neprijatnosti zavisi od faktora, nabrojanih u uvodu. Kada je opterećenje operatera u pitanju, tu treba imati u vidu i razne načine rada. Sa stanovišta zaštite na radu najteže je unošenje podataka, jer je to najstatičklji, najmonotoniji, najnezanimljiviji posao, koji iziskuje veliku preciznost i koncentraciju. Kreativni poslovi su raznovrsniji i živahniji. Kod nekih se računar koristi samo kao povremeno pomagalo i ljudi su različiti – jednima, recimo, više odgovara kreativan, a drugima rutinski posao. Motivisanost pojedinca za određeni posao u velikoj meri utiče na njegovo opšte stanje.

Iziskustva pokazuju da su najčešći, pa i najopravdaniji razlozi za nezadovoljstvo i teškoće u radu računarom povezani sa pogrešno uređenim radnim mestom, a kod starijih naprava i sa lošom mašinskom i neprijatnom programskom opremom; dodatni uzroci tegoba jesu nedovoljna osposobljenost i slaba motivisanost radnika, a ako su u pitanju osobe u godinama, tu je i otpor prema radu računarom. Zapažamo takođe da su radnici često pogođeni ili uvređeni ako im se objasni da uzrok njihovih tegoba nisu razna opasna zračenja, već mnogo banalniji razlozi i da im je zajednički imenitelj loše uređeno radno mesto.

Kako treba radno mesto da izgleda da rad računarom ne bi bio neprijatan i štetan?

Osnovno je da prostorija ispunjava zahteve u pogledu odgovarajuće toplote, svetlosti i tišine, odnosno zaštite od buke. To znači temperaturu vazduha oko 23 stepena Celzijusa, relativnu vlagu oko 50% i brzinu kretanja vazduha do 0,2 m/s. Za teže, složenije, kreativne poslove buka ne bi smela da prelazi 55 dB(A), a za rutinske 65 dB(A). Kao ilustraciju recimo da jačina mirnog govora iznosi oko 65 dB(A). Manja odstupanja od tih zahteva obično nisu neposredno štetna po zdravlje, ali mogu da budu neprijatna, čoveku smetaju u radu i pogoršavaju mu opšte stanje.

U radu računarom najvažnija je rasveta na radnom mestu, pa ćemo je detaljnije obraditi.

Prirodna ili veštačka rasveta mora da obezbedi:

- zadovoljavajuću i ravnomernu osvetljenost prostorije (osvetljenost se meri luksima – lx),
- zadovoljavajuću kontrast i izostrežnost znakova na ekranu (kontrast se izražava relativnom razlikom između osvetljenosti znaka i njegove pozadine),
- svetlost ne sme da treperi (to podjednako važi za osvetljenost prostorije i za znake na ekranu),
- boje u prostoriji moraju da budu usklađene.

Naročito veliki zahtev predstavlja uslov da radnici za ekranima imaju vidni kontakt s prirodnim spoljnom okolinom, a da je radno mesto dobro osvetljeno po svakom vremenu i u svako doba dana, mešovitim prirodnim i veštačkom svetlošću.

Kako treba da izgleda računarsko radno mesto?

Mišljenja da osvetljenje radnog mesta treba da bude što jače velika je zabluda. Sasvim je dovoljno ako su radne površine osvetljene jačinom svetlosti od 250 do najviše 500 lx. Ako je svetlost jača, kontrast znakova na ekranu slabi, jer pozadina ekrana postaje suviše svetla (to važi za ekrane sa svetlim znacima na tamnoj pozadini), a i zidovi, tavanica i druge svetle površine u prostoriji tako su snažno osvetljene da počinju da se ogledaju u ekranu.

Najvažniji su osvetljenja i odnos osvetljenosti u vidnom polju radnika. Svetlost predstavlja jedinu fotometrijsku količinu koju oko neposredno registruje. Svetlost (meri se kandela, odnosno svećama) utoliko je veća što je sjajnije rasveteno telo i što je osvetljena površina manja. Oko se samo prilagođava raznim svetlostima, a za to mu treba izvesno vreme. Što je veća razlika u svetlosti, oku će za prilagođavanje trebati više vremena i napora. Zato je razumljivo što se oko zamara usled neprestanog prelazanja pogleda sa svetle okoline na tamni ekran računara i nazad, a taj zamor utoliko jači što je veća razlika između svetlosti okoline i ekrana. Da se oči ne bi previše naprezale, odnos svetlosti u prostoriji i na radnom mestu, za računarom tamnog ekrana i svetlih znakova, mora da bude u granicama, šematski prikazanim na slici 1.

Uslov da svetlosti površine u vidnom polju operatera budu odgovarajuće usklađene znači da prozori i druge svetle površine moraju da budu sa strane, a nikako iza ekrana ili leđa operatera. Ako su mu svetle površine iza leđa, ogledaju se na površini ekrana, što smanjuje kontrast među znacima i pozadnom i operater znake teže raspoznaje.

Da bi radno mesto bilo odgovarajuće osvetljeno treba imati u vidu još nekoliko dodatnih uslova:

- mašinska oprema mora da bude svetla, bez tamnih ivica oko ekrana na jedinicama za prikazivanje ili oko tastature;
- zidovi, tavanica, nameštaj, radni sto i pult za čitanje moraju da budu svetli;
- rasvetna tela na tavanici moraju da budu zasenčena zrcalnim rasterom, kako se ne ogledala u ekranu;
- prostorija treba da ima prozore okrenute ka severu, a ako to nije moguće, prozori treba da budu zaštićeni zavesama, recimo, belim zavesama vertikalnih lamela, kako bi se po svakom vremenu i u svako doba mogao sprečiti prodor direktne sunčane svetlosti u radnu prostoriju.

Zahtev za izjednačenjem svetlosti u vidnom polju operatera ne znači da arhitekta ne treba prostoriju prijetno da opremi i osveži pastelnim nijansama boja, raznim dodacima u boji itd. Operatera ne sme da ometa kretanje drugih ljudi predmeta iza ekrana. Zato iza ekrana ne smeju da budu nikakvi prolazi, transportni putevi ili predmeti u pokretu.



Računar iziskuje bezličan dijalog prema precizno utvrđenim pravilima i zato operator posle izvesnog vremena dobija osećaj usamljenosti. Veoma velike prostorije, ispunjene velikim brojem radnika, taj osećaj samo pojačavaju, pa se zato treba pridržavati pravila da u računarskoj radnoj prostoriji treba da bude više radnika, ali ipak ne više od šest.

Ne treba isticati da prostorija mora da bude dovoljno velika i pregledna, kako bi se rad i protok gradiva organizovali miran i logičan način.

Za zdrav rad je od izuzetnog značaja i kancelarijska oprema, ali njoj se u praksi ne posvećuje dužna pažnja. Kancelarijska oprema mora da bude praktična i funkcionalna, pri čemu prvo pravilo glasi da radna površina mora da bude dovoljno velika. U praksi obično nije tako. Računar, ekran i tastatura zauzimaju dosta mesta, a obično se stavljaju na već inače pretrpan pisali sto, dok se radni materijali drže na kolenima, obližnjem radijatoru, prozorskoj polici ili na podulj. U takvim uslovima neizbežno je raditi iskrivljene kičme i vrata, pa su i tegobe sa skeletom i mišićima neizbežne – prvo kao prolazni bolovi, a kasnije i kao ozbiljna bolest.

Veoma koristan dodatak na stolu je polica, na koju se stavlja ekran, dok ispod nje ostaje slobodna radna površina. Radna površina mora da bude glatka i svetla, ali mat, da ne odbija svetlost. Sto mora da bude dovoljno dubok, tako da ispred tastature ostaje 10 centimetara površine, na kojoj ruka nalaze oslonac. Prilična radna površina ostaje slobodna ako računar ne stoji na pisacem stolu, odnosno ako odaberete varijantu u vidu stuba, smeštenog pored pisacag stola.

U kancelarijsku opremu obavezno spada i ergonomski oblikovana stolica sa podešivim sedištem i naslonom, sa točkicama i bez naslona za ruke.

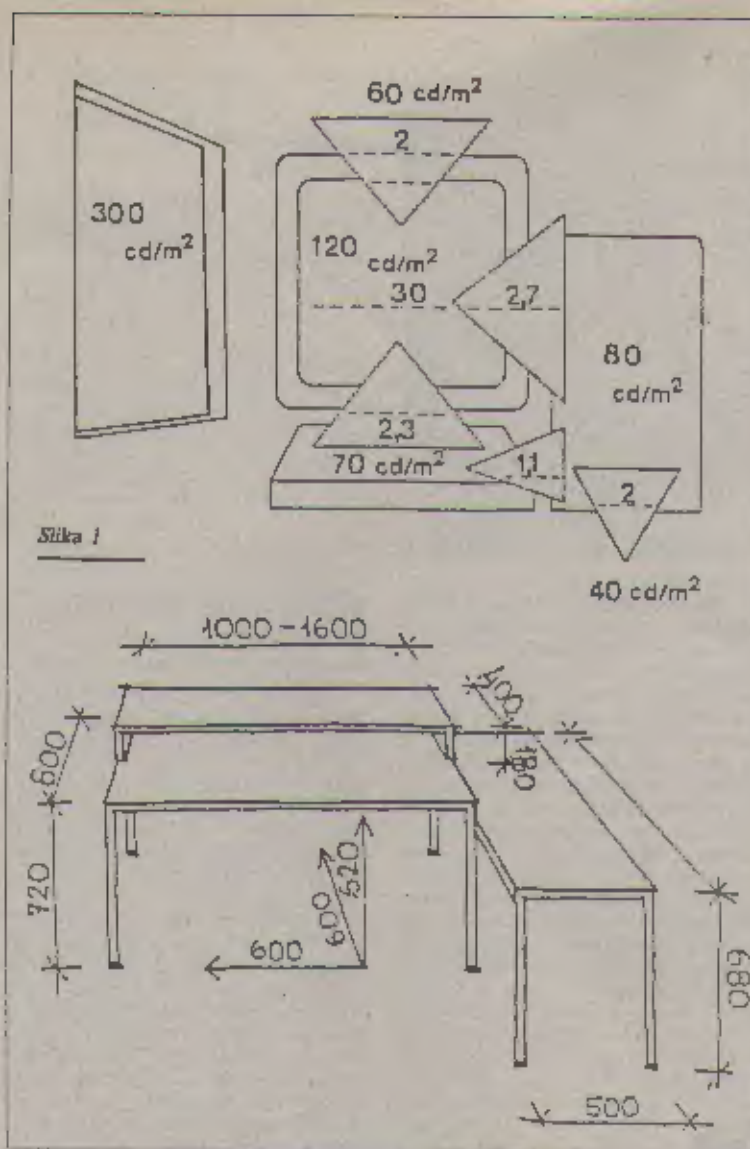
Personalni računar traži još dva specifična komada opreme: stočić za štampač i oštacki pult.

Oslonac za noge pominjemo samo usput, jer ide uz svako kancelarijsko radno mesto.

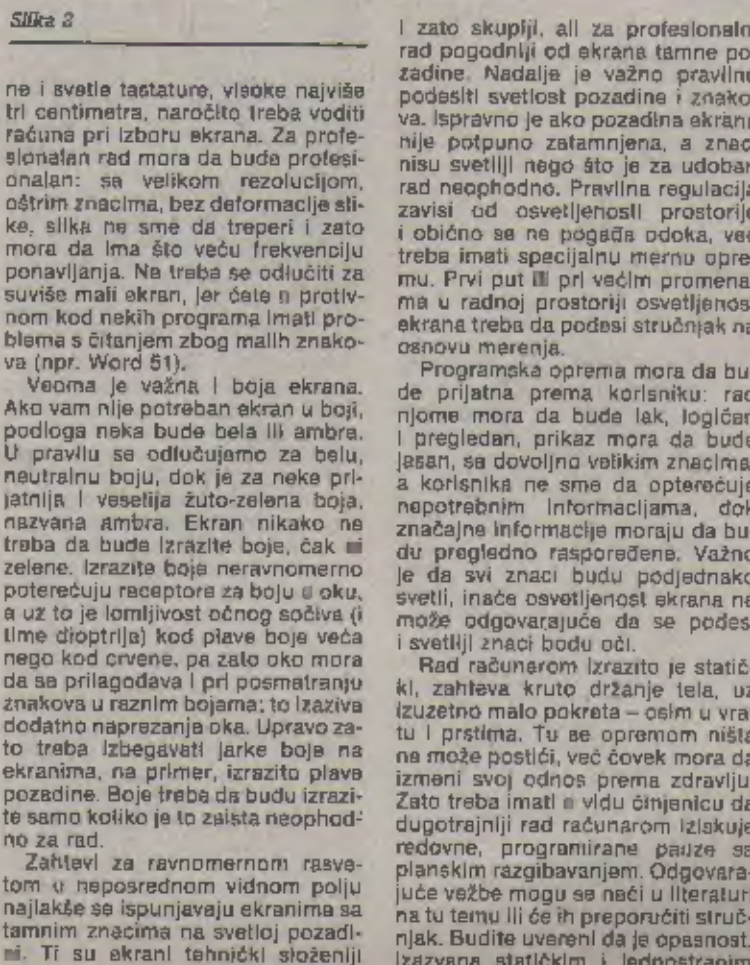
Raspored stola i stočića za štampač preporučenih dimenzija šematski je prikazan na slici 2. Odgovarajuća visina radnog stola obično iznosi 72 centimetra.

Sve radne površine moraju da budu stabilne. naročito upozoravamo na nedostatke malih stočića sa klimavom policom za tastaturu na izvlačenje. Stočić, najčešće na točkicama, na kojoj ima mesta samo za računar, ekran i štampač, ima opravdanja samo ako za pravi sto zaista nema mesta ili ako za računarnom radite samo povremeno i sklapanje ga kad vam ne treba. Treba izbegavati i stolove sa iscečenom i produbljenom policom za tastaturu, jer iscečeni deo stola nameće mesto tastature na radnoj površini; ako je izrez u sredini, ubrzo ćete uvideti da nemate mesta za rad ni levo ni desno od njega.

Iako su personalni računari danas pravilu ergonomski dobro oblikovani, imaju svetla kućišta, pregled-



Slika 1



Slika 2

na i svetle tastature, visoke najviše tri centimetra, naročito treba voditi računa pri izboru ekrana. Za profesionalan rad mora da bude profesionalan: sa velikom rezolucijom, oštrim znacima, bez deformacije slike, silka ne sme da treperi i zato mora da ima što veću frekvenciju ponavljanja. Ne treba se odlučiti za suviše mali ekran, jer ćete u protivnom kod nekih programa imati problema s čitanjem zbog malih znakova (npr. Word 51).

Veoma je važna i boja ekrana. Ako vam nije potreban ekran u boji, podloga neka bude bela ili ambr. U pravilu se odlučujemo za belu, neutralnu boju, dok je za neke prijatnija i veselija žuto-zelena boja, nazvana ambra. Ekran nikako ne treba da bude izrazite boje, čak ni zelene. Izrazite boje neravnomerno poterećuju receptora za boju u oku, a uz to je lomljivost očnog sočiva (i limbe dioptrija) kod plave boje veća nego kod crvene, pa zato oko mora da se prilagođava i pri posmatranju znakova u raznim bojama; to izaziva dodatno naprezanje oka. Upravo zato treba izbegavati jarke boje na ekranima, na primer, izrazite plave pozadine. Boje treba da budu izrazite samo koliko je to zaista neophodno za rad.

Zahtev za ravnomernom rasvetom u neposrednom vidnom polju najlakše se ispunjavaju ekranima sa tamnim znacima na svetloj pozadini. Ti su ekran i tehnički složeniji

i zato skuplji, ali za profesionalni rad pogodniji od ekrana tamne pozadine. Nadalje je važno pravilno podešiti svetlost pozadine i znakova. Ispravno je ako pozadina ekrana nije potpuno zatamnjena, a značnici nisu svetliji nego što je za udoban rad neophodno. Pravilna regulacija zavisi od osvetljenosti prostorije i obično se ne pogađa odoka, već treba imati specijalnu mernu opremu. Prvi put ili pri većim promenama u radnoj prostoriji osvetljenost ekrana treba da podešiti stručnjak na osnovu merenja.

Programska oprema mora da bude prijatna prema korisniku: rad njome mora da bude lak, logičan i pregledan, prikaz mora da bude jasan, sa dovoljno velikim znacima, a korisnika ne sme da opterećuje nepotrebnim informacijama, dok značajne informacije moraju da budu pregledno raspoređene. Važno je da svi znaci budu podjednako svetli, inače osvetljenost ekrana ne može odgovarajuće da se podeš i svetliji znaci budu oči.

Rad računarnom izrazito je statički, zahteva kruto držanje tela, uz izuzetno malo pokreta – osim u vratu i prstima. Tu se opremom ništa ne može postići, već čovek mora da izmeni svoj odnos prema zdravlju. Zato treba imati u vidu činjenicu da dugotrajniji rad računarnom izazuje redovne, programirane pauze sa planskim razgibavanjem. Odgovarajuće vežbe mogu se naći u literaturi na tu temu ili će ih preporučiti stručnjak. Budite uvereni da je opasnost, izazvana statičkim i jednostranim

opterećenjima i neprirodnim držanjem tela, mnogo ozbiljnija od štetne koju navodno prčinjavaju razna »zračenja«.

Šta, zapravo, zrači računar?

Da li je u radu za ekranom računara potreban filter na tom ekranu ili ne? Korišćenje filtera obično se pogrešno obrazlaže zaštitom od zračenja koje ekran emituje.

Činjenica je da ekranska jedinica sa katodnom cevi emituje razna fizikalno merljiva zračenja, od rendgenskog, preko nisko i visokofrekventnih električnih i magnetnih polja, do infracrvene, vidljive i ultravioletne svetlosti. Naravno, treba imati u vidu intenzitet tih zračenja u apsolutnim vrednostima i njihov značaj u odnosu na zračenja kojima je čovek izložen u svakodnevnom životu. Rendgensko zračenje nastaje usled kočenja elektronskog snopa na fluorescentnom sloju u katodnoj cevi. Ekran je premazan fluorescentnim premazom tako da zasvetli na mestu, pogođenom elektronskim snopom. Rendgensko zračenje utoliko je jače što je jači taj elektronski snop (struja elektrona) i što je veći anodni napon u cevi. Od napona u cevi zavisi i čvrstina, odnosno prodornost rendgenskih zraka.

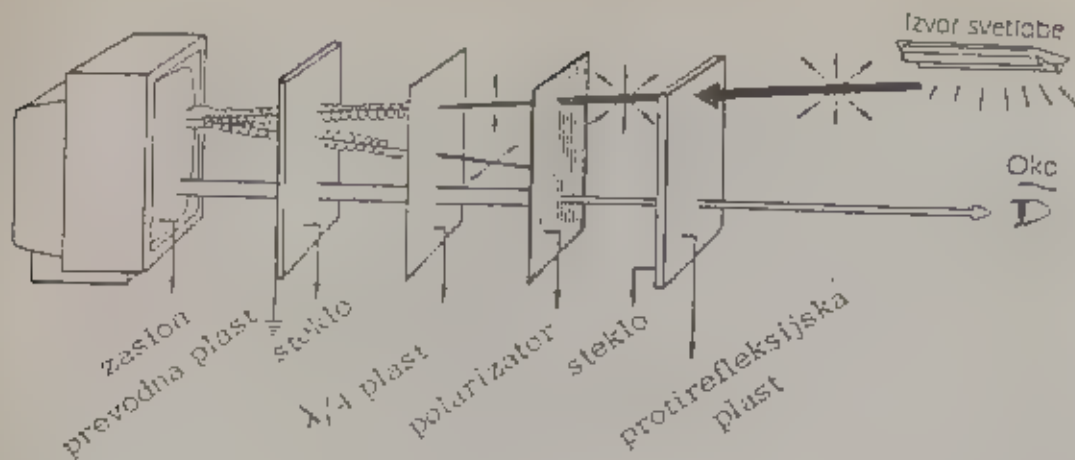
Anodni napon cevi obično se označava na ekranskoj jedinici. Kod cevi koje rade naponom ispod 20 kV, rendgensko zračenje je tako slabo da se praktično potpuno apsorbuje u staklu ekrana i zato je svaki strah izlisan. Prodornije i jače zračenje može da se očekuje kod većih ekrana i ekrana »boji«, ako rade naponom iznad 20 kV.

U svakodnevnom životu čovek je izložen i izvorima jonizujućeg zračenja: kosmičkim zracima, zračenju prirodno radioaktivnih elemenata i zračenju radona u vazduhu. Dodatno je izložen zračenju veštačkih izvora, u koje spadaju i računarski ekran, zatim, zračenju usled zagađenja životne sredine kao posledice eksplozija atomskih bombi, nuklearnih centrala, proizvodnje veštačkih radionuklida i zračenju pri lekarskim pregledima.

Detaljna merenja zračenja najrazličitijih tipova ekrana pokazala su da operator pred ekranom s katodnom cevi – puno radno vreme tokom godinu dana dobija otprilike hiljadu puta manju dozu nego što je iznosilo prirodno zračenje pre Černobila.

U cilju umirenja može da posluži i propis o zaštiti od zračenja, koji zahteva strogu proveru ekranskih jedinica, a dokaz je izjava na vidnom mestu na kućištu da jonizujuće zračenje ne prelazi granice bezbednosti za operatera. Uredaji bez takve izjave ne mogu da se prodaju ni u inostranstvu, ni kod nas.

Merenja svih vrsta nejonizujućih elektromagnetnih zračenja dokazala su da električna ili magnetna polja oko ekranske jedinice nisu ništa veća od onih kojima smo izloženi u prirodi, kod kuće ili pri obavljanju drugih sličnih poslova. Jedina značajna električna kolibna može da bude



Slika 4

elektrostatično polje pred ekranom. Zbog tog polja ekran privlači više prašine nego drugi predmeti u prostoriji. Iz istog razloga više prašina se taloži i na ekranu kućnog televizora i površinama oko njega. Ali, i statično električno polje ne prelazi granice za koje do sada nisu utvrđeni nikakvi štetni uticaji po čoveka.

Računar stvara elektromagnetna polja niskih i srednjih frekvencija, ali merenja pokazuju da su i ta polja daleko ispod granice koja bi, kako dosadašnja iskustva pokazuju, mogla štetno da deluje na čoveka.

jas spektra. Zato je u načelu zaista moguće da ekran zrači jedan deo energije i u ultravioletnom području, ali na mestu operatera može da se očekuje samo dugotrajni deo ultravioletnog spektra, koji staklo ekrana još propušta. Međutim, u praksi je malo verovatno da ćete naći na jednobojni ekran sa takvim fluorescentnim premazom da osjetno zrači u ultravioletnom području spektra. To je verovatnije kod ekrana u boji.

Merenja su pokazala da je intenzitet zračenja ekrana u obilžnjem ultravioletnom području najmanje hiljadu puta manji od prirodnog ultra-

Filteri – više štete nego koristi

Zašto se onda takvi filteri uopšte proizvode i na veliko prodaju? Prvo zbog lakovernih kupaca, uverenih da će se na taj način zaštititi od štetnih zračenja. Zanimljivo je da su ozbiljni prodavci toga svesni, što ćemo uvideti ako njihove reklame čitamo pažljivo i bez želje unapred da u njima nađemo uveravanje o zaštiti od zračenja.

Kakve koristi obećavaju proizvođači filtera? Odgovor je sledeći: otklanjanje smetnji u vidu refleksa (odbleska) na površini ekrana. Samo usput pominju da filter sprečava širenje statičnog električnog polja u okolinu ekrana, što je svakako korisno, jer ekran ne treba tako često brisati. Takođe pominju da filter zaдржава širenje elektromagnetnih talasa u okolinu, što je možda tačno. Ali od toga nema nikakve praktične koristi, jer je to zračenje tako slabo da za radnika nema značaja.

Razni filteri na razne načine smanjuju odblesak na površini ekrana. Najjednostavniji filteri su mrežice ili neutralno obojeno (zatamnjeno) staklo. Te dve vrste filtera u pravilu provode i električnu struju i zato takvi filteri, ako su uzemljeni, sprečavaju širenje elektrostatičnog polja u bližu okolinu. Kako takav filter svojom obojenošću, odnosno apsorpcijom svetlosti smanjuje odblesak, šematski prikazuje slika 1.

Tu se vidi da filter tu svetlost koja pravi smetnje (označena A) prvo oslabi pre nego što padne na ekran, a zatim još jednom, kad se odbijena svetlost vraća ko posmatraču. Svetlost znakova koje zrači ekran (označena B) filter oslabi samo jednom, pri prolasku od ekrana ka posmatraču. Zahvaljujući tome poboljšava se odnos intenziteta A/B1 između zračene (korisne) i odbijene (loše) svetlosti, čime se pojačava kontrast znakova u odnosu na pozadinu ekrana. Filter, dakle, pojačava kontrast u nepovoljnim uslovima osvetljenja u kojima operateru smeta odblesak svetlih predmeta na ekranu. Međutim, treba znati da korekcija filterom uopšte ne bi bila potrebna da je prostorija odgovarajuće osvetljena, odnosno ako se drugim merama spreči odblesak rasvjetnih tela ili prozora, kao posledica neodgovarajuće rasvete u prostoriji ili pogrešnog položaja ekrana.

Kad su vrste filtera u pitanju, mrežice ne preporučujemo, jer smanju-

ju rezoluciju i lako se prljaju. Najbolji su višeslojni polarizacioni filteri, koji su zbog složene tehnologije opravdano neuporedivo skuplji. Šematski prikaz delovanja polarizacionog filtera nalazi se na slici 4.

Polarizacioni filter na veoma zanimljiv način polarizuje tu ometajuću svetlost, prvo linearno, a zatim još kružno. Svetlost koja se odbija od ekrana kružno je polarizovana, ali s izmenjenom fazom. Odbijenu svetlost filter ponovo polarizuje od kružne u linearnu, ali pošto joj je prethodno već izmenio fazu, sada je više ne propušta. Zato polarizacioni filter u potpunosti sprečava ometajući odblesak svetlosti sa ekrana, ako ugao pod kojim svetlost pada nije suviše mali i ako zanemarmmo odblesak od samog filtera, ali taj je veoma mali.

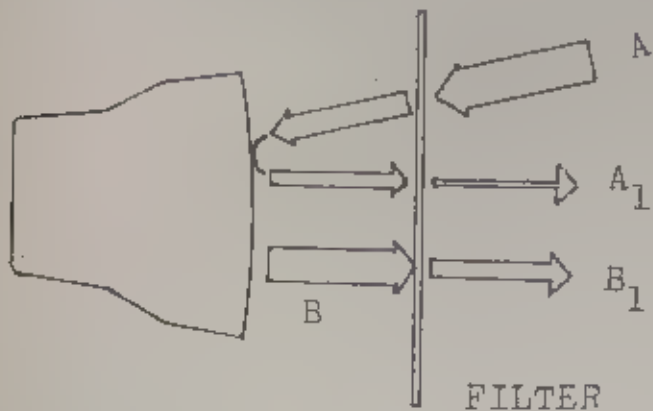
Zbog smanjenog odbleska pojačava se kontrast. Letimično posmatrao, slika izgleda oštrije, ali na tu filter nema zapravo nikakvog uticaja – pre suprotno, ako se upotrebimrežica.

Filter, dakle, zaista više ili manje efikasno smanjuje odblesak i time doprinosi boljem kontrastu slike, ali pogoršava vidljivost, jer ekran postane tamniji. Time se remeti jedno od osnovnih načela pravilne osvetljenosti, kao što je prikazano na slici 1. Zato je filter na jednoj strani konstantan, a na drugoj šatan.

Konačan zaključak glasi: korišćenje filtera u cilju zaštite od zračenja nije opravdano, dok je korišćenje filtera u cilju sprečavanja odbleska opravdano samo ako se odblesak ne može sprečiti nikakvim drugim merama. Tu treba napomenuti da odblesak, pored toga što zamara oči, posredno utiče na pogrešno držanje tela operatera i time i na ozbiljna oštećenja lokomotornog aparata. Zato je sprečavanje odbleska jedan od najvažnijih zadataka pravilnog uređivanja računarskog radnog mesta.

Na kraju pomenimo još jednu primedbu korisnika personalnih računara, da se u nekim zemljama stručnicima navodno preporučuje da izbegavaju rad računarima? Osnovni razlog nije štetno zračenje, već u uvodu pomenute, izuzetno nepovoljna kombinacija niza štetnih uticaja: tu se naročito ističu stresna situacije kao posledica loše organizacije rada, i zahtevi kojima radnik nije dorastao. Sve to treba u trudnoj i svakako izbegavati.

Dr. Primoz Gaspar je vanredni profesor na Vloj tehnološkoj školi za bezbednost u Ljubljani.



Slika 3

Svaki zagrejeni predmet emituje infracrveno ili toplотно zračenje, pa i računar sa svojom ekranskom jedinicom. Ali, svi mi znamo da je toplотно zračenje računara neznatno u odnosu na zračenje sunca, peći ili radijatora. Tako možemo sa sigurnošću da kažemo da ne može da bude štetno.

Zadatak ekranske jedinice je da zrači vidljivu svetlost u obliku što prepoznatljivijih simbola. Pošto smo u svetlošnim uslovima za dobro prepoznavanje znakova na ekranu već poverili, u zračenju vidljive svetlosti ovde neće biti ponovo reči.

Ima pojedinaца koji su nepoverljivi prema ultravioletnom zračenju. Fluorescentni premaz na ekranu ne emituje svetlost sa precizno utvrđenom talasnom dužinom, pa ni jednu jasno određenu boju, već zračena svetlost zauzima užu ili širu pa-

violetnog zračenja sunca napolju ili bar deset puta manji nego u prostorijama sa fluorescentnom rasvetom. Ozbiljno strahovanje od izlaganja ultravioletnoj svetlosti nije na mestu. Svi takođe znamo da nam koža preplane ako duže boravimo na suncu, dok od rada za računarom, bez obzira koliko intenzivno bio, još niko nije preplanuo! Tačno je svakako, da je štetna i prirodna ultravioletna svetlost, jer izaziva rak kože. Na to lekari sve glasnije upozoravaju, naročito sada kad su oštećeni ozonski slojevi u gornjim delovima atmosfere.

Na osnovi intenzivnih studija i merenja možemo, dakle, da konstatujemo: danas nema naučno potvrđenog dokaza da su zračenja kojima je izložen radnik ispred ekranske jedinice sa istodnom čevi zaista štetna ili da su jača od zračenja kojima su izloženi drugi radnici. To takođe znači da nikakvi dodatni filteri ispred ekrana u cilju zaštite operatera nisu potrebni.

Nikada poštom

BORIS HORVAT

Ne isplati se. Prema trenutno važećem carinskom zakonu programska oprema se (osim sistemskih programa u velike računare) ne carini. jer se smatra intelektualnom svojinom (isto kao knjiga). Carini se samo vrednost medija, to jest disketa, koje staju najviše dolara svaka, što znači da su carinski troškovi za prosečan programski paket sa pet disketa, trebalo da budu najviše 30 (trideset) dolara (priručnici su kao knjige oslobođeni carina). Međutim, stvari nisu tako jednostavne. Zato moramo da objasnimo uvodno izlaganje.

Programske pakete obično poručujemo u nekoj od većih američkih trgovina (Programmer's Warehouse, Programmer's Paradise, Programmer's Connection, The Programmer's Shop i sl.), čije oglase možemo da pronađemo u svim američkim računarskim revijama. Sve ove trgovine omogućavaju veoma povoljno naruđivanje telefonom i plaćanje kreditnim karticama a osim toga, imaju najpovoljnije cene (i do 50 odsto niže nego sami proizvođači programске opreme) jer poručuju od proizvođača na veliko i uzivaju naravno odgovarajuće popuste. Cene u njihovim oglašima stvarno su najniže. Međutim, to samo tako izgleda. Sve trgovine bez izuzetka (i na izričito drukčiji zahtev) šalju pošiljke, preko jedne od ekspresnih kurirskih službi koje rade u Evropu zalista dostavljaju za tri dana ali to, razumljivo naplaćuju (od oko 50 USD za manje pakete, kao što su Norton Guides, do 150 USD za veće - recimo neki C++); ove troškove možemo od oka da ocenimo na 50 USD po kilogramu. Problemi počinju baš zbog takvog načina slanja jer kad pošiljka stigne avionom na Brnik, preuzima je jedna od špeditorija koja za primaoca obavija carinska formalnosti. Prema carinskom zakonu svaka pošiljka koja vredni više od 10 USD, mora da se carini posredovano špedicijom. Ovo pravilo se doduše, ne poštuje sasvim dosledno ako pošiljka stigne običnom poštom dok ekspresna kurirska pošta špediciju ne može da izbegne. Evo i mogućnosti pošiljke kurirskom službi Federal Express preuzima Transjug Rijeka za DHL brine ŽG Feršped Ljubljana, a UPS Extremely Urgent brine InterEuropa Koper.

Posredovanje špedicije čine, pored plaćanja carine, sledeći troškovi: taksene marke teton, poštarina, carinska deklaracija provizija za položenu gotovinu, 5-procentni porez za JNA - sve to iznosi 100 do 150 dolara, što je najmanji deo troškova. Drugi tajni udeo uzima špediter za posredovanje taj udeo trenutno iznosi od 444 (InterEuropa), preko 500 (Transjug), do 600 dolara (ŽG Feršped). Treći deo su skladišni troškovi,

utvrđeni od 80 do 100 dinara (skladišna manipulacija, otvaranje i zatvaranje pošiljke, prijava robe carinarnici), a uz to se za svaki dan plaća ležarinu od 10 do 20 dinara, zavisno od veličine (težine) paketa, s tim što uobičajeni postupak traje 5 do 10 dana.

Naravno, ne smemo da zaboravimo carinu. Tek tu počinje pravi problem. Carina automatski uzima kao carinsku osnovu vrednost paketa prema fakturi, ne mareći što je u paketu programska oprema, za koju bi prema carinskom zakonu smela da obračuna samo vrednost disketa. U stvari carina odnosno špediter u deklaraciji pravilno napiše da su u pitanju «snimljeni nosioči za zvučne i druge fenomene - snimljena disketa u korisničkim softverom» (tarifni broj 85 24 80101) ali uprkos tome, kao carinska osnova uzima se puna vrednost paketa. Ako imate sreće carina ili špediter dodaje carinskoj osnovi i troškove poštarine, jer za to ima osnovu u istom carinskom zakonu. Ta «areča» može skupiti da vas košta, jer vrednost poštarine kod kurirske službe dostiže skoro vrednost same pošiljke. Kao primer uzet ćemo program Windows 3.0, koji staja 99 USD, a poštarina 80 USD. Carina je doduše, spremna da poštuje zakon da se carini samo medij - disketa. Ali, naravno zahteva originalni račun, gde je cena programskog paketa podeljena na cenu diskete i cenu programa. Naravno, slobodno zaboravite da bi naka od navadenih softverskih kuća bila spremna da vam ispostavi takav račun (provereno u dva slučaja). Sa jednim carinikom na Brniku uspeši smo doduše, da se složimo da je dovoljno ako špediter podeli vrednost na dva dela (disketa - program), mada smo do sada uspeši da ubedimo samo jednog špeditera (Transjug) da to učini. Da ne bismo Transjug samo hvalili recimo i da je to prema našem iskustvu, jedini špediter koji u deklaraciji carinskoj osnovi dodaje i troškove pravoza (naknada za prevoz) a uz to naplaćuje troškove dostavljanja Brnika u Ljubljani (210 dinara), bez obzira što kurirske službe u cenu usluge već uračunavaju lično dostavljanje do kupca. Na carinsku osnovu platiti ćete oko 16,5 odsto carine (preduzeće) nešto manje od 40 odsto (građani). Pa, mnogo sreće!

Šta učiniti? Ako programski paket već ne možete da poručite drukčije nego preko kurirske službe morate znati da možete potpuno legalno da izbegnete nepotrebno plaćanje carine na program. Naravno, morate biti dovoljno uporni u objašnjavanju zakona, kako kod špeditera, tako i kod carinika. Odlazak na Brnik može se smatrati suvlasnim mada mnogo pomaže. Treba upozoriti da se odugovlačenjem postupka povećavaju skladišni troškovi.

Ako pošiljka dođe normalnom (avionskom) poštom, carini se na



carinskoj pošti na železničkoj stanici u Ljubljani (naravno važi za Ljubljani). Ako je vrednost paketa ispod 30 USD nema carinskih dažbina a ako je veća na vrednost medija (diskete) treba platiti oko 60 odsto dažbina (na Brniku oko 40 odsto). Za preduzeća i radne organizacije pošiljke, naravno, odlazi na carinjenje u Javna skladišta, gde je postupak sličan opisanom (špediter itd.). Redakcija «Mog mikro» mora uvek da moli i da ubeđuje carinike na carinskoj pošti da je u pitanju programska oprema koju dobija kao uzorko za test i da te programe ne šalju dalje u Javna skladišta (Ovde carinicima moramo da odamo priznanje, jer je do sada sve bilo uspešno, mada i dalje ne možemo da se oslobodimo utiska da nam čine posebnu uslugu odnosno da gledaju kroz prste.) Još malo kritike poštarima na carinskoj pošti. Kad smo prošlog proleća dobili paket poštom običnom poštom izgubilo se na oko mesec dana uprkos kompletnoj adresi (urednik, revija) poštarima su poslali obaveštenje na «DELO Titova 35».

To obaveštenje, naravno, nije pronašlo put do redakcije «Mog mikro». Plaćimo se da se u međuvremenu još jedna pošiljka zgubila na sličan način proizvođač programске opreme iz SAD ljubazno nam je poslao paket običnom avionskom poštom (da bi izbegao pomenute probleme sa špedicijama na Brniku), ali ni posle dva meseca još ga nismo dobili!

S tim u vezi postavljamo javno pitanje. Zašto se pošiljke programске opreme, poslate kurirskom poštom, carine na Brniku posredovanjem špedicije a pošiljke poslate običnom poštom bez posredovanja

loterija na Brniku softver nije prtljag.

špedicije?

Postoji još jedna mogućnost, stodošto proverena i legalna. Inače ne bismo navodili. Neka vam neko kupi programski paket u SAD, platiće istu cenu kao i vi (ako slučajno ne živi u istoj saveznoj državi u kojoj je sedište trgovine, morate da platite porez od 3 do 6 odsto a za troškove kurirskog dostavljanja platiće 10 do 20 USD, zavisno od težine pošiljke) i u SAD sve programske pakete šalju kurirskom poštom. Iz paketa treba da uzme diskete i da vam ih pošalje običnom avionskom poštom i nešto tvrđem kovertu i drugi deo pošiljke (priručnika) treba da pošalje kao štampane stvari. Troškovi za sve zajedno biće daljih 10 do 20 dolara. Kroz dve sedmice naći ćete diskete u svom poštanskom sandučetu a priručnike ćete podići na centralnoj pošti u Čopovoj ulici (i ovo važi za Ljubljani) gde ćete možda morati i platiti nekoliko desetina dinara za troškove carinske pošte (otvaranje paketa). Treba upozoriti da ovaj put nije sasvim pouzdan jer ako se pošiljka izgubi nemate mnogo izgleda da je ikad vidite (pošiljke preko kurirske službe su osigurane).

Ovom prilikom nećemo se baviti drukčijim načinima kupovine (lična kupovina u Inostranstvu, kupovina od jugoslovenskih distributera, kupovina u našim trgovinama), a piratstva smo se ionako već svi odrekli, zar ne?

P.S. Kakvo je vaše iskustvo sa pošiljkama programске opreme poštom? Pišite redakciji «Mog mikro» s naponom na «Uvoz programске opreme», naročito ako ste probali bolje nego mi.

DISKETE 3.50" i 5.25"

— MAXELL — BASF — NASHUA — TDK — 3M — SONY —
— PRECISION — NO NAME —

- Nudimo kvalitetne diskete za svakoga po trenutno najpovoljnijim cijenama u Jugoslaviji
- Narudžbe šaljemo isti dan
- Na veće narudžbe dajemo popust, a na manje poklon
- Još danas zovite i uverite se u raznovrsnost naše ponude i prihvatljiva cijena

TIGER, d.o.o.

poduzeće za promet roba i usluga

Matoševa 2, SESVETE

tel. 041/202-200

Lego za hakere

Tekst i foto:
ROBERT SRAKA

Prošli put smo «proučili» tvrde diskeve i ostalu gvozdariju. Uputimo se, sa tim znanjem, prodavcu i kupimo delove za računar ili već sastavljeni računar. Naravno da u prvom slučaju treba računar još sastaviti, pa je zato krajnje vreme da kažemo, kako se to radi. Postupak nije tako komplikovan, iako zbog mnoštva različitih uređaja (prvenstveno tvrdih diskova) ne postoje jednostavni i jednoznačni saveti. Za sastavljanje je potreban odvračač, nekoliko disketa sa «trenutku prikladnim» programima i malo spretnosti. Osim toga, da bi računar stvarno radio, ne sme imati maler sa kupljenim delovima.

Kućiče

Kao prvo raspakujte kućiče. Ono obično ima, u prilogu, kutijicu sa različitim zavrtnjima (obično ■ je malo više), shemu za podešavanje cifara na displeju za prikazivanje frekvencije, metalne pločice sa kojima se zatvaraju neiskorišćeni otvori za kartice i još neke stvari. Po otvaranju kućiče, kao prvo proverite da li je ispravljač dobro pričvršćen. Ako nije montiran, a obično nije, pričvrstite ■ na predviđeno mesto. Zatim pogledajte kako se ■ kućiče pričvršćuju memorijske jedinice (disk, disketna jedinica...). Ovdje razlikujemo dva slučaja: kod nekih kućiča se disk i disketna jedinica pričvršćuju direktno na kućiče, a kod drugih na priložene nosače. Nosači su plastični ili metalni. Po njihovom pričvršćenju na disk (to ćemo raditi kasnije), stavite ih ■ zleb ■ kućiču i disk ugurajte u ležište. Metalni nosači se obično zavrtnjima pričvršćuju na kućiče sa prednje strane, a plastični se sami uglavljaju u pravilan položaj. Iako još nećete montirati disk ili disketnu jedinicu, važno je da pogledate kako se to radi. Kod kućiča sa nosačima za spoljne memorijske jedinice nema nikakvih teškoća, pa zato možete početi sa montažom osnovne ploče. Kod kućiča, kod kojih se ove jedinice pričvršćuju na kućiče mogu se pojaviti teškoće. Neka (prvenstveno mala vertikalna kućiča) napravljena su tako, da se po pričvršćenju matične ploče, ne može vi- ■ pričvrstiti 3,5-colska jedinica. U takva kućiča treba prvo pritrditi 3,5-colsku jedinicu, pa zatim montirati osnovnu ploču.

Kad ste kućiče dobro pregledali, zaplajte gde su sve priključene žice koje više iz kućiče. To su obično priključci za svetleće diode koje signalizuju da je računar uključen, da tvrdi disk radi i da li je uključen turbo režim rada, zatim priključak za bravu za tastaturu, priključak za zvučnik i priključak za «turbo» ta-



Slika 1. Testiranje računara IPC multisys.

ster. Nekada treba, neku od nabrojanih stvari, pričvrstiti i na kućiče. Skoro uvek treba pričvrstiti i zvučnik. Ako je displej za prikazivanje frekvencije montiran na kućiče koji se može podešavati samo sa unutrašnje strane, onda ga treba podešiti pre nego što počnete sa montažom da bi izbegli akrobacije sa odvračačem kad je računar već spreman za rad. Naravno, prvo treba pregledati dokumentaciju koja se dobija uz osnovnu ploču i potražiti podatak na kojim frekvencijama ploča radi. Obično ■ dva frekvencije, za računare AT ili 386 ■ MHz i deklarirana frekvencija ploče (12, 16, 20, 25 ili 33 MHz). Ali to nije pravilo. Neke ploče mogu da rade na više frekvencija, ali kod njih se treba odlučiti samo za jednu ili dve koja će se prikazivati na displeju za frekvenciju.

Osnovna ploča

Latimo se sada osnovne ploče! Čipovi na osnovnoj ploči su osjetljivi.

Slika 2. Rastavljen računar u sredini su oba Intelova čipova, gore levo su memorijski moduli SIM, na vrhu je procesorska ploča.



vi. Na nema se, prvenstveno hodanjem po sintetičkom tepihu ili nekim drugim trenjam, nakupi električni naboj. Kada neoprezno dotaknete čip, naboj se preko njega prazni i on se oštećuje. Ovaj naboj se zove «statički elektricitet». Zato se treba, pre početka rada sa bilo kojom štampanom pločom i memorijskim čipovima, razelektriti. To činimo hvatanjem vodovodne česme ili nečega drugog što ■ uzemljeno. Na osnovnu ploču stavite prvo memorijske čipove (ako nisu stavljeni). Pri stavljanju čipova obratite pažnju na njihovu orijentaciju. Tačka ili polukružni zarez na čipovima mora biti na istoj strani kao i zarez na podnožju. Isto tako obratite pažnju ■ orijentaciju i kod memorijskih modula SIM. Memorijske čipove, kao module, pritisnite ■ podnožje do kraja. Na osnovnoj ploči se je i niz skakača (jumpera) koji se lakše podešavaju pre montaže. Ako imate neku sporiju osnovnu ploču AT, podešite stavljenom količinu memorije. To se, pri oživljavanju računara sa štampanom pločom NEAT, radi sof-

tversko. Premeštanje skakača je potrebno i u slučaju, kada računar ima grafičku karticu ■ boji. ■ tom slučaju treba u dokumentaciji, koja ■ dobija uz osnovnu štampanu ploču, potražiti koji skakač služi za određivanje tipa grafičke kartice. Drugi važan skakač je onaj koji oživljava matematički koprocesor (naravno ako postoji). On je na pristupačnom mestu isto kao i podnožje za koprocesor, pa se zato može podešiti kasnije. Ako je vaše kućiče takvo, da je po montaži osnovne ploče nepristupačan skakač ■ podnožje koprocesora, onda koprocesor montirajte pre montaže osnovne ploče. Opet obratite pažnju na orijentaciju. Koprocesori 8087 i 80287 imaju na ivici zarez ili tačku, a 80387SX i 80387 imaju oštećen jedan ugao.

Kada ■ RAM (a možda i koprocesor) stavljen, pogledajte rupa za pričvršćivanje osnovne štampane ploče na kućiče. Štampana ploča se pričvršćuje plastičnim rasplinjačima i sa dva ili tri zavrtnja. Pogledajte koji se prerezi na kućiču podudaraju sa montažnim rupama na štampanoj ploči i u njih stavite plastične rasplinjače. Na mesta na kućiču gde će se štampana ploča pričvrstiti zavrtnje kratke metalne rasplinjače. Zatim osnovnu štampanu ploču pažljivo stavite u kućiče. Pogledajte, da li ■ svi plastični rasplinjači pravilno u ležištima i zavrtnje dva ■ ili zavrtnja u metalne rasplinjače. Sa tim je najteži deo posla završen. Zatim priključite sve svetleće diode, zvučnik, taster «turbo» i bravu za tastaturu. Neke osnovne štampane ploče nemaju priključak za taster «turbo». Kod njih taj taster ne treba nigde priključivati. Frekvenciju rada možete zatim menjati samo preko tastature (obično kombinacijom tastera Ctrl/Alt/+ i Ctrl/Alt/-).

Proverite da slučajno ispravljač nije priključen na mrežni napon. Iz ispravljača više dva priključka ■ napajanje osnovne štampane ploče (prvi je označen sa P8, a drugi sa P9). Tačno pregledajte dokumentaciju i oba priključka priključite na štampanu ploču. Pri tome treba biti veoma oprezan! Ako na priključcima ne postoje oznake, onda se razlikuju po bojama žica. Za uzemljenje su obično crni, a za napon od 5 V crveni. Nepravilno priključeni priključci (npr. zamenjeni) mogu prouzrokovati uništenje matične ploče. U podnožje stavite grafičku karticu. Ako se radi o VGA kartici ili sličnoj, proverite da li podešeni tip monitora, na vašoj kartici, odgovara vašem monitoru. Za monohromatsku karticu hercules nije potrebno ništa učiniti.

Sada je računar spreman za prvi test. Zatvorite kućiče, priključite monitor i tastaturu, uključite monitor i računar (tim redosledom). Ako je sve u redu, na ekranu se ispisuje poruka proizvođača BIOS-a, a zatim računar protestira da konfiguracija

ne odgovara. U tom slučaju ugasiite računar i kada računar ne pokazuje znakove života, ne treba praviti paniku. Greška je verovatno banalna, isključite monitor, isključite kabl za napajanje, nepravilno priključen glavni prekidač na kućištu (ovaj će verovatno prouzrokovati kratak spoj u kućnoj električnoj instalaciji), monitor nije priključen na grafičku karticu ili je intenzitet monitora podešen na minimum.

Montaža diskova

Ugasiite računar i montirajte tvrdi disk i disketnu jedinicu, ako to zbog nezgrapnog kućišta nije trebalo uraditi još pre montaže osnovne ploče. Ako imate više disketnih jedinica ili tvrdih diskova treba svima osim poslednjem odstraniti pakete sa zaključnim otpornostima (to je opisano u dokumentaciji koja se dobija uz disk). Pakete sa zaključnim otpornostima treba da ima samo poslednja jedinica (pošto ćete verovatno imati u računaru samo jedan disk, možete na taj podatak zaboraviti). Uz tvrdi disk i disketnu jedinicu dobija se nekoliko pljosnatih kablova, obično dva ili tri. Broj zavrtki od upravljača. Za upravljače MFMM i RLL potreban je za disketnu jedinicu jedan široki kabl, a za tvrdi disk jedan uži. Za upravljače AT/IDE i ostale obično je za disketnu jedinicu potreban jedan široki kabl, a za tvrdi disk jedan malo širi. Za upravljače MFMM i RLL su široki kablovi jednake širine i sa jednakim konektorima, ali se ne smeju zameniti! Kabl za disketnu jedinicu ima između oba priključka za disketnu jedinicu više izukršanih žica nego kabl za tvrdi disk. Možete za priključenje su na upravljaču i disketnoj jedinici označene brojevima. Žica u kablju koja odgovara broju 1, je drugačije boje od ostalih, pa se zato skoro ne bi smelo desiti da ga pogrešno uključite. Konektor na kablju za 3,5-colsku disketnu jedinicu je drugačiji konektor za 5,25-colsku disketnu jedinicu. Zato kada kupujete 3,5-colsku disketnu jedinicu proverite da li kabl ima dodatni konektor za tu jedinicu odnosno da li ima dodatnu pločicu preko koje se priključuje na kabl za 5,25-colsku disketnu jedinicu. Kod 3,5-colskih disketnih jedinica postoji zub sa kojim se konektor podudara sa disketnom jedinicom, često na pogrešnoj strani. Ako tako onda ga osetite (pri tome pazite na kabl!).

Priključite sve kablove. Ako imate samo po jedan disk i disketnu jedinicu, priključite ih na drugi, to jest na poslednji konektor na kablju. Konektorom na upravljaču spojite svetleću diodu koja označava rad diska. Upravljač stavite u prorez, što bliže napojnom delu (ili ilme disku). Disk i disketnu jedinicu priključite na napajanje. Napajanje ne priključite na silu. Ako konektor ne možete da priključite, verovatno pokušavate da ga priključite naopako!

Na kraju, u podnožje stavite karticu sa serijskim i paralelnim veznikom. Ako u računaru postoji grafička kartica koja nema paralelni veznik, treba pre stavljanja podasiti skakače na kartici sa paralelnim

veznikom. Obično je, u sistemu (LPT2), paralelni veznik na kartici konfigurisan kao drugi veznik, jer veznik na grafičkoj kartici konfigurisan kao prvi. Ako grafička kartica nema veznika, tada treba veznik na vezničkoj kartici konfigurisati kao prvi (LPT1). Pre zatvaranja računara, pametno je da sve žice uvijete i pričvrstite na kućište. To je najlakše učiniti izolovanom žicom sa kojom je bio uvezan kabl za napajanje. Pljosnate kablove savijte i fiksirajte gumicom. Takođe pričvrstite neupotrebljene kablove za napajanje disketnih jedinica i diskova da slučajno ne bi pali na matičnu ploču (što bi se moglo završiti tragično). Kada ste sve to obavili, možete računar zatvoriti, zavrtiti zavrtinje na kućištu i sve je spremno za

Oživljavanje računara

To može biti veoma težak zadatak ukoliko bilo koja komponenta ne



Slika 3. Processorska ploča mušunjs SX: dole desno je Intelov mikroprocesor 386 SX, nad njim je promer za koprocesor

funkcionisati kako bi morala. Takav problematičan računar namučiće i stare vukove u sastavljanju računara. Na sreću, većih poteškoća obično nema pogotovo ako ste kupili standardne komponente. Nekim poteškoćama koje se mogu javiti govorilićemo kasnije kod opisa sastavljanja ili ispitivanja testnog računara. Ovde ćemo se baviti samo oživljavanjem zdravih računara.

Kada je računar konačno sastavljen ponovo ga uključite. Računar po pregledu memorije, zaprepašeno izveštava da konfiguracija ne odgovara stvarnom stanju. Obično zahteva da startujete program Setup koji je u BIOS-u na osnovnoj ploči. Starije osnovne ploče to nisu poznavale, pa je bilo potrebno podšavanje specijalnim programom BIOS je u računarima sa procesorom 386 ili 386SX obično proizvod firme AMI i solidno je izrađen. Program Setup se u BIOS-u AMI uvek aktivira pri startu računara, kad se pritisne taster «Delete». Kod ostalih BIOS-a je start drugačiji obično se to izvodi tasterima «Esc» i «F1» ili kombinacijom «Ctrl-Alt-Esc». Pri startu Setup-a, ispisuje se trenutna konfiguracija. To treba promeniti tako da odgovara našoj konfiguraciji. Pri tome treba navesti i tip tvrdog diska. Pomoćice vam podaci o kapacitetu, cilindrima i broju glava koji su navedeni u dokumentaciji (ili u prošlom broju Mog Mikra, ako je disk naveden u tabeli). Ako u si-

stemskoj tabeli nema diska sa istim karakteristikama kao što je vaš, potražite drugi najbliži u toj tabeli.

Obavezno mora imati jednak broj glava, a ostali podaci mogu biti različiti. Ako na taj način odredite disk, gubi se razlika između kapaciteta našeg diska i navedenog u sistemskoj tabeli računara. Tip diska se kod nekih računara može odrediti u polju «user type». U njemu se navode stvarni parametri diska. Nažalost retko se uz disk dobija bilo kakva dokumentacija to prouzrokuje teškoće. Ako kupujete manje poznati disk, ne zaboravite da upitate prodavca za njegove fizičke karakteristike! Kada je u računaru upravljač tvrdog diska sa vlastitim BIOS-om, tada se u Setupu određuje ona vrednost koja su zapisane u dokumentaciji za upravljač (nezavisno od stvarnog diska). Upravljaču dajte stvarne podatke u disku, kada startujete program u njegovom BIOS-u.

Obično se kapacitet memorije i ostali detalji (slična memorija, način preplitanja memorije itd.) ne mogu podasiti standardnim Setupom. Zato koristite prošireni Setup ili neki poseban program (npr. SetNeat) koji se dobija na disketi uz osnovnu ploču. Računar se resetira čim se potvrde izmene koje ste uneli u Setup. Računar će, po završenom testiranju svojih komponenti, pokušati da unese operacioni sistem sa diskete. Ponudite mu ga. Po završenom unošenju, za rad se može osposobiti i tvrdi disk.

Priprema tvrdog diska

Delimo je na više operacija: predfORMATIRANJE, razdeljivanje i formatiranje. Za predfORMATIRANJE treba tražiti podatke po celoj particiji (logičkom disku) a ne u ostalim particijama, onda je to istina. Pošto su podaci koji se ovako ili onako, odjednom koriste, obično ili prilično zbijeni ili rastreseni po većim particijama i pošto postoje dosta brzi programi sa kojima se, s vremena na vreme, mogu pospremiti datoteke, onda argumenti u korist particija nisu više tako jaki. Sam sam protiv njih jer kod skoro pune particije treba celo vreme brinuti a tome koliko je prostora još na raspolaganju. Za neku aplikaciju koja se s vremenom jako širi treba često delove podataka ili neke druge programe koji smetaju saiti iz jedne particije a drugi (iako vas to po drugoj strani čuva od prevellke aljkavosti). Pogledajmo primer: ako disk NEC 3142 priključite na upravljač RLL i koristite DOS 3.3 onda morate napraviti bar tri particije ukoliko želite da ko-

stupnog softvera. Za većinu diskova se, pri kupovini, dobija program (npr. Disk Manager) koji sam brine za sve. Program se unese odradit tip diska (sada oznakom) i pričekava se da program završi svoj posao. U međuvremenu program nas upita samo o preplitanju sektora i kapacitetu particija (ako ih želite).

Predformatiranje

Predformatiranje je najlakše prepuštiti nekom programu. Na početku treba samo navesti oštećene sektore i vaš posao je završen. Ako program ne zna da potraži najprikladnije preplitanje sektora, vrednost postavite na deklarisanu vrednost vaš kontroler (obično 1 ili 2). Ako taj podatak ne znate dobro je da disk pripremite sa preplitanjem 1:3 testnim programom i proverite brzinu diska i disk ponovo pripremite sa manjim preplitanjem. Na taj način se utvrđuje najbolje preplitanje. Kod novijih diskova i kontrolera nikada nije potrebno preplitanje iznad 1:3.

Predformatiranje diska se, kod nekih osnovnih ploča, vrši testnim programom u BIOS-u koji se startuje na sličan način kao i standardni Setup. Takav program imaju neki AMI-ovi BIOS-i. Osim predfORMATIRANJA, zna da potraži i optimalno preplitanje. Disk se na sličan način predfORMATIRA i iz DOS-a. Ali nijedna od ove dve varijante neće, u mnogim slučajevima, raditi zbog nestandardnih upravljača (za BIOS i DOS su standardni samo upravljači MFMM). Obično se disk može, kod nestandardnog upravljača, predfORMATIRATI iz upravljačkog BIOS-a (o tome smo govorili u prošlom broju). Neki diskovi prvenstveno diskovi AT/IDE su, o krasna stvar, već predfORMATIRANI. Za njih vam ne treba gubiti vreme sa operacijom koja može da traje i nekoliko desetina minuta.

Particioniranje i formatiranje

Već smo rekli, kad treba napraviti particije. Neki ipak zagovaraju podelu na particije i kad nisu potrebne jer su pristupi do datoteka brži. Ako tražite podatke po celoj particiji (logičkom disku) a ne u ostalim particijama, onda je to istina. Pošto su podaci koji se ovako ili onako, odjednom koriste, obično ili prilično zbijeni ili rastreseni po većim particijama i pošto postoje dosta brzi programi sa kojima se, s vremena na vreme, mogu pospremiti datoteke, onda argumenti u korist particija nisu više tako jaki. Sam sam protiv njih jer kod skoro pune particije treba celo vreme brinuti a tome koliko je prostora još na raspolaganju. Za neku aplikaciju koja se s vremenom jako širi treba često delove podataka ili neke druge programe koji smetaju saiti iz jedne particije a drugi (iako vas to po drugoj strani čuva od prevellke aljkavosti). Pogledajmo primer: ako disk NEC 3142 priključite na upravljač RLL i koristite DOS 3.3 onda morate napraviti bar tri particije ukoliko želite da ko-

ristite 65.2 Mb raspoloživog prostora. Ako napravite tri jednake particije i na svakoj imate po dva Mb slobodna, onda ćete biti veoma akučani. Mnogi programi zapisuju na disk velike međudatoteka i za takve je dva Mb slobodnog prostora jednovredno premalo. Ako imate samo jednu particiju, uvek bi bilo i na toliko zauzetoj disku na raspolaganju šest Mb prostora, što obično zadovoljava i najpreždrljiviji program.

Bilo kako bilo, ako particija ne napravi neki program, napravite ga u DOS-ovim naredbama FDISK. To je naredba koja korisnika, za razliku od većine ostalih naredbi, vodi po menijima. Sa njom se određuje broj i veličina particija kao i aktivna particija. Ako želite da na disku imate samo jednu particiju, aktivirajte FDISK, odredite ceo disk kao particiju i recite da je particija aktivna. Računar aktivira DOS samo iz aktivne particije. Ako particiju ne aktivirate, računari pri svakom aktiviranju zahteva sistemsku disketu.

Particije (odn. particija) se formiraju naredbom Format. Na aktivnu particiju (obično logički disk C:) prenesite sistemske datoteke (naredbom Format c: /s). Ako DOS nije ukrali, to uređuje originalni instalacijski program Install. Kada ste to uradili, prekopirajte na disk i ostali deo operacionog sistema. Prema uputstvu iz priručnika napišite za operacioni sistem datoteku autoexec.bat i config.sys.

Poslednje operacije

Kada je računar osposobljen za normalan rad, pogledajte još jednom, da li su sve svetleće diode pravilno priključene i da li rade onako kako bi morale. Proverite da li možete ključem zaključati tastaturu i da li radi taster »reset«. Proverite jednim programom, npr. Landmark Speed, kolika je frekvencija na kojoj računari radi. Ovde moram još jednom da naglasim da računari može da radi samo na jednoj od standardnih frekvencija (tj. 10, 12, 15, 20, 25 i 33 MHz). Ako test Landmark pokaže neku drugu vrednost, to ide na račun stanja čekanja. Landmark daje pravilne vrednosti računaru u jednim stanjem čekanja. Kod računara bez stanja čekanja ili sa preplitanjem memorije može pokazati do 30 procenata višu vrednost od stvarne. Ako je neku od gore nabrojanih stvari treba promeniti, ugasite računari i pokušajte da grešku popravite.

Pri ponovnom uključanju može se desiti da računari opet zahteva podešavanje konfiguracije. To se dešava onda, kad se akumulator na osnovnoj ploči još nije napunio, pa je računari zato zaboravio konfiguraciju, kao što je sat i datum. Zato tek sastavljeni računari ne treba ugasiti odmah po sastavljanju. Ako računari koriste kao komad nameštaja, što se nažalost i kod nas već događa, posle izvesnog vremena može da izgubi podatke o konfiguraciji (zato što se akumulator retko puni). Iznad su ponekad sa tim akumulatorima problemi, pa zato osnovne ploče imaju priključak i za spoljnu bateriju.

Optimizacija rada

Kada računari konačno radi, možete učiniti još dosta za poboljšanje njegovih svojstava. O tome je, u našoj reviji, bilo već nekoliko članaka. Pre nego što počnete da razmišljate o programima koji na razne načine ubrzavaju neku operaciju, pokušajte da što više »izvučete« iz hardvera. Na mnogim novijim osnovnim štampanim pločama mogu se pomoću programa (prošireni Setup u BIOS-u, Program SetNeat ili slični) promeniti mnogi parametri štampane ploče, kao što su stanje čekanja, preplitanje memorije, kapacitet memorije, prenos memorije između 640 i 1024 K na mesto iznad 1024 K, uključivanje proširene memorije EMS, brzina sabirnice, stanje čekanja na sabirnicama, uključivanje senčne memorije i još neke stvari. Mogućnosti je toliko da se ne mogu ukratko opisati, a osim toga veoma zavise od tipa osnovne ploče. Zato ćemo o optimizaciji više govoriti za svaki testni računari posebno (u sledećem, definitivno poslednjem delu ove serije). Treba znati da se menjanjem parametara ploče mogu postići znatna poboljšanja, pogotovo ako imate u računaru dovoljno memorije. Zašto ih ne bi iskoristili, ionako ste ih platili!

IPC multisys

IPC multisys je veoma zanimljiv računari. Izgrađen je malo drugačije od uobičajenih SX-a. Procesor kod sistema multisys nije na matičnoj ploči već na posebnoj kartici. To omogućava kasnije nadgrađivanje računara drugom procesorskom karticom bez menjanja bilo čega u računaru. Naravno, kupičeto ga sa sastavljenog i sa garancijom »on site«.



Slika 5. Kartica za proširenje, za priključenje osam terminala na računari 386 SX (ili više) koji radi pod Unixom.

te». Takođe je zanimljivo samo kućiste, jer je napravljeno tako da se može koristiti kao »baby« ili stani tower. Na stranicama nema zavrtnja, a natpisi i displej za frekvenciju stoje ukoso, tako da je svejedno kako kućiste stoji. Šteta je samo da prekidač za uključanje nije na prednjoj strani. Dalje od prvog utiska pomoćnice nam odvraća.

Osnovna ploča

Pre par godina kada se i mali računari izrađivali modularnoj

Slika 4. IPC multisys sa procesorskom pločom i grafičkom karticom teeng mega eva/1024.

tehnici, nazvali su sabirnicu, na koju su se priključivale sve ploče (procesorska, memorijska itd.), matična ploča. Taj naziv je kod ličnih računara promenio svoje značenje, jer ovde naziv matična ploča znači ploča sa mikroprocesorom. I dalje je, u kontakte na matičnoj ploči, bilo potrebno stavljati kontrolere za pogon disketnih jedinica i tvrdih diskova, serijske i paralelne veznike te grafičke kartice. Stvar se kod multisysa opet obrnula. Matična ploča nema procesora. Na njoj dominiraju dva Intelova kontrolera 82230 i 82231. Oba rade na frekvenciji do 20 MHz. Na krajnje levoj strani ploče (najudaljenije od ispravljača) nalazi se dugačko podnožje za procesorsku ploču. Pored njega je osnaest podnožja za SIM memorijske module. U priloženoj dokumentaciji je opisano samo osam podnožja koja dozvoljavaju brzu memoriju od 512 K do 8 Mb. Brzina memorije može biti 100ns ili 120ns. U prvom slučaju možete birati između tri načina organizacije i pristupa brzaj memoriji: običan, sa preplitanjem i sa četvorstrukim preplitanjem. Poslednji način je najbrži, ali za njega treba imati na raspolaganju pune fondove. Ostalih osam podnožja je verovatno predviđeno za procesorske ploče sa prvim procesorom 386.

Na matičnoj ploči je još kontroler AT/IDE (host adapter) za dva tvrda diska i kontroler za disketne jedinice, izgrađen oko koprocesora WD37C65B (koprocesori su sve inteligentna kola koja glavni procesor oslobađaju od dodatnih radnji: ono, šta u vezi sa pečelima obično nazivamo koprocesor u stvari je matematički koprocesor, a u računaru postoji više koprocesora koji brinu za ostale radnje). Naravno da taj kontroler zahteva odgovarajući tvrdi disk. Ako hoćemo da damo običan disk sa odgovarajućim kontrolerom, kontroler na matičnoj ploči se jednostavno isključuje skidanjem kapiće sa skakača (jumpera) isto tako se može uključiti ili isključiti i kontroler za disketnu jedinicu. Proverite sam kako radi ova opcija sa običnim kontrolerom MFM i Seagateovim diskom ST4096. Sve je radilo prema očekivanju, a ST4096 je bio tako strašno bučan kao i obično (naravno da to nema veze sa pločom). To da se adapter za tvrdi disk još uvek nalazi na osnovnoj ploči je ugodno, jer prenos podataka ne postoji međusabirnicama između kartice i os-

novne ploče, interesovalo me je, kako dobro to radi. Ispitivanje sam vršio sa Seagateovim ST1126A. Prvi put sam ga priključio na host adapter na matičnoj ploči, a zatim na spoljni host adapter. Upotrebio sam Core test. Disk je, sa spoljnim adapterom, postigao brzinu prenosa od oko 800 K/s, vreme pristupa 16 milisekundi i vreme za prelaz sa traga na trag 5 milisekundi. Rezultati sa adapterom na štampanoj ploči su bili 520 K/s, 14.2 milisekunde i 1.2 milisekunde. Druga dva vremena su osetno bolja, a razlog zašto je Core zapisao manju brzinu prenosa, ne znam. Na oko mi se nije činilo da je disk bio sporiji, pre bih rekao brži. Na matičnoj ploči su još jedan paralelni i dva serijska veznika. Konektori za njih i konektori za tvrdi disk i disketnu jedinicu su na desnoj strani ploče, porad ispravljača. Konektori su veoma pristupačni, a isto važi i za skakače sa kojima se kontroler i veznici uključuju. Samo sam, za podešavanje adresa i prekida za paralelne i serijske veznike, morao izvući karticu iz zadnje utičnice.

Ploča je snabdevena AMI-ovim BIOS-om koji je prilagođen ploči IPC. Ploča se podešava normalnim Setupom. Na raspolažanju je lokacija memorije između 640 i 1024 K, uključivanje ili isključivanje senčne (shadow) memorije, i to je skoro sve. Pošto na ploči nema čipova NE-AT, oštećeni smo za simpatično podešavanje parametara ploče i uopšte se ne može desiti da se ploča zaključa. Processorska ploča je pune dužine i manje više je prazna. Na njoj su samo procesor i kristal koji brine za njegov ritam, veliki Intelov čip 82335, podnožje za matematički koprocessor, nekoliko pojačavača linija i logičkih kapija. Na njemu su još dva skakača kojima se podešava način pristupa do brze memorije

Grafička kartica i monitor

Grafička kartica, u računaru, je Tsangova MegaEva/1024. To je kartica VGA rezolucije 1024 x 768 u 256 boja sa prepilanjem. Taj broj boja zauzima 640 Mb memorije. Kartica je izgrađena oko novog Tsangovog čipa T4000 koji se od prethodnika (T3000) razlikuje prvenstveno po brzini. Zajedno sa karticom dobijaju se još tri diskete i biblioteka sa uputstvima. Na prvaj su standardni pogoni za AutoCad, Lotus, Gem i ostalo, što već spada u tu grupu. Osim testnih programa i programa za podešavanje grafičkog režima na disketi je još i editor za ekranne znakove. Tako se naši krovčići mogu brzo nacrtati. Najviše me je obradovao sadržaj druge diskete, jer je puna pogona za MS Windows 3.0. Ostale kartice na tržištu, uglavnom još nemaju pogone za novu verziju «prozora». Pogone sam naravno odmah isprobao. Sve je radilo kako trebal. Na trećoj disketi su tri veoma simpatične slike formata 1024 x 768 x 256. Testni monitor je bio IPC VDVGA14, to je uobičajeni monitor tipa «double scan». Dobro je radio u svim režimima VGA (uključujući i 1024 x 768)

osim sa proširenima koji omogućavaju ispisivanje 132 znaka u redu. Ovi režimi mu nisu baš išli najbolje od elektroda.

Operativni sistem

Operativni sistem je neophodan deo računarske opreme. Dobro je da, pri kupovini računara, bar njega kupite. Nažalost kod nas se još uvek dešava da čak i preduzeca kupuju računare bez operacionog sistema, pa ga zatim prekopiiraju iz nekog drugog računara. Verovatno bi i kod nas morali, kao u Nemačkoj, propisati da se svaki računar mora obavezno prodavati zajedno sa operativnim sistemom.

Na računaru IPC multsys ispitao sam četiri operaciona sistema: MS-DOS 4.01, XENIX 386, DR-DOS 3.41 i DR-DOS 5.0. Najmanje mogu da kažem o najvećem među njima, XENIX-u. On je već bio instaliran na računaru kad sam ga dobio na testiranje. Njemu spada još obimna gomila knjiga, koja mi je uzela svu volju do toga da malo preturam po meni nepoznatim stvarima. Zato sam ispitao samo neke stvari iz priručnika, a one su radile. Rad je sa ovim operativnim sistemom brz, naravno dosta drugačiji od rada sa DOS-om. Za neki pravi test su potrebni programi koji rade pod XENIX-om. Pošto se bez njih nije moglo dalje, zamenio sam disk i jedan po jedan instalisao gore navedene DOS-ove. Obe verzije DR-DOS-a bile su IPC-ove, to jest sa prilagodnicima IPC. Ni nova verzija DR-DOS-a nije odstranila neke nedostatke starije verzije (3.41), prvenstveno je rad sa diskom sporiji od MS-DOS-a za nekih 10 procenata. To se dosta primjećuje pri radu sa većim datotekama. Inače nova verzija ima mnogo lepih osobina (vidi prošli broj MM) koje bi tu sporost mogle provagnuti. Nažalost, ova verzija ima prilične teškoće sa paralelnim veznikom. Neki programi koji do sada nisu nikada otkazali, nisu pod novom verzijom uopšte radili. Pri ispisivanju bitne slike na ekran, štampač je s vremenom na vreme potpuno zablokirao. To je radio i kada sam rezultat programa snimio na datoteku i ovu kasnije poslao na štampač sa narednjom copy'om. Istu datoteku sam uspešno odštampao kako pod DR-DOS-om 3.41 tako i pod MS-DOS-om 4.01. Narednja copy'om radi, pri slanju datoteke na štampač, nekoliko puta sporije nego u staroj verziji. Ovak operacioni sistem sam ispitao još na dva druga računara i rezultat je bio isti.

Iz toga se može izvući da IPC multsys dobro radi sa različitim operativnim sistemima, ali nije sigurno da će raditi ono šta se od njih očekuje.

Naravno, isto pravilo važi i za računare koje ste sami sastavili – ponekad nije za sve kriv hardver. Nažalost nisam mogao da ispitam kako radi XENIX kada se na njega priključiti još nekoliko terminala.

P. S.: Autor se zahvaljuje preduzeću 7L iz Murške Sobote na pozajmici računara IPC multsys i navedenog softvera.

NOVO !!! PC software



kompjuterski
rječnik - editor

ugrađen
adresar i
podsjetnik

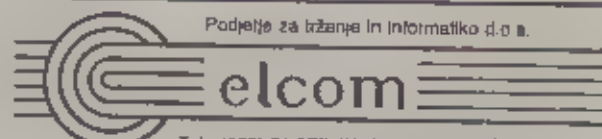
englesko - hrvatski
njemačko - hrvatski
svaki po 15000 riječi

program
rezidentan

plaćanje pouzećem, narudžbe telefonom i faxom

Kaštela tours

58212 Kaštel Sućurac, Putaljski put bb
telefon 058/657-333; telefax 058/657-101



Podjetje za tržanje Informatiko d.o.o.

Tel.: (066) 24-977, 23-685 - Telefax: (066) 24-981
JLA 5 66000 KOPER YUGOSLAVIA

- INSTALACIJE I ODRŽAVANJE VIŠEKORISNIČKIH SISTEMA UNIX, PC-MOS, ...
- PRODAJA I SERVISIRANJE RAČUNARSKE OPREME I SASTAVNIH DELOVA ZA PERSONALNE RAČUNARE
- OBRAZOVANJE KORISNIKA
- IZRADA PROGRAMSKE OPREME PO NARUDŽBINI
- RAČUNARSKE KOMUNIKACIJE

**U FEBRUARU VAM NUDIMO
ŠTAMPAČE PO POSEBNO POVOLJNIM
CENAMA, POZOVITE NAS NA TEL:
(066) 24-977**

NOVOST!

SPASITE, ŠTO SE MOŽE
- UPOTREBITE NEPREKIDNO
NAPAJANJE NA KARTICI
»ACCUCARD«

**ELCOM, KAD OD RAČUNARA
ZAHTEVATE NAJVIŠE!**

AT-once protiv AT-speeda

TOMAŽ ISKRA
Foto: IGOR MODIC

Već je prošlo nekoliko godina otkad je nastao prvi MS-DOS emulator za računare atari ST. Prisetimo se: to je bio softverski emulator s imenom PC_DITTO. Prava korist od takvog emulatora, zbog njegove nemoguće sporosti (Nortonov faktor 0.3), nije se mogla očekivati. Taj američki softverski proizvođač ipak nije bio sasvim beskoristan: pokazao je put u razvoju sistema ST i probudilo zanimanje brojnih korisnika Atarijevih računara. To je već bilo dovoljno da se razvojem i poboljšanjima MS-DOS emulatora počelo baviti sve više ljudi.

Juče XT, danas AT

Prvi, koji su u drukčijim rešenjem problema emulacije prešli u ređiju na dela, bili su Nemci. Tako su se pre nešto više od godine dana istovremeno pojavila čak dva hardverska emulatora IBM XT. Oba su bila napravljena oko procesora NEC-V30 na 8 MHz. Emulator PC-SPEED bio je pločica štampanog kola koju je trebalo zalemiti na Motorolin procesor MC68000. SUPERCHARGER je bio skoro samostalan XT s vlastitim RAM-om, koji je trebalo priključiti na atari ST preko DMA interfejsa. Po konceptiji emulatori su se razlikovali u prvom redu po tome da je za delovanje PC-speeda na početku bilo potrebno žrtvovati nešto vremena za ugradnju i rizikovati da sve zajedno možda ipak neće tako lepo raditi. Tako bi, pored bačenog novca, «unakazivanje» ST-a bilo uzaludno, jer iskustava s tim proizvođačem u stvari nije bilo. Kod Superchargera kirurških zahvata nije bilo. Ipak ubrzo se pokazalo da je PC-speed veoma dobro zamišljen. Njegove glavne prednosti bile su:

- na stolu nije zahtevao dodatni prostor
- nije zahtevao dodatno napajanje i nije zauzimao dodatne interfejsa na atariju ST.

U prvom redu, PC-speed je bio

AT-once

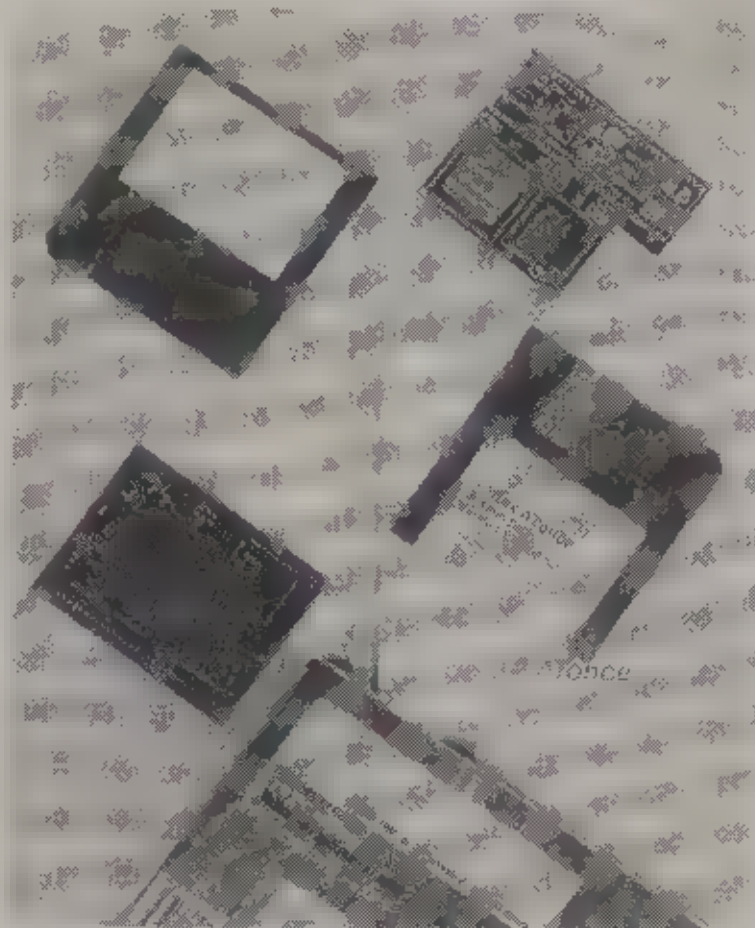
BI-System Information, Advanced Edition 4.90, (C) Copy 1987-88, Peter Norton

Computer Name: IBM AT
Operating System: HUP 3.10
Built-in BIOS dated: Wednesday, 23 August 1990
Main Processor: Intel 80286 Serial Ports: 2
Co-Processor: None Parallel Ports: 1
Video Display Adapter: Color/Graphics CGA
Current Video Mode: Text, 80 x 25 Black and White
Available Disk Drives: 4; A: - D:

DOS version: 704 K-bytes of memory:
77 K-bytes used by DOS and resident programs
627 K-bytes available for application programs
A search for active memory finds:
640 K-bytes main memory (at hex 0000-0A000)
128 K-bytes display memory (at hex 0A000-0C300)
2 640 K-bytes extended memory (at hex 10000-30000)

Computing Index (CI), relative to IBM/AT: 6.7
Disk Index (DI), relative to IBM/AT: Not computed, No drive specified.

Performance Index (PI), relative to IBM/AT: Not computed.



izuzetno pouzdan. Poslednja verzija programa (1.41), koja omogućava rad emulatora, bila je kompletna. Sve stare greške bile su odstranjene.

Godinu dana kasnije napravili su dva novija emulatora. Ovaj put to su bili AT emulatori: AT-SPEED (nastao od uspešnog PC-speeda) firme SACK ELECTRONIC GMBH i AT-ONCE firma VORTEX. Testiranja obaju emulatora omogućilo nam je zagrebačko preduzeće MICRO COMPUTING, Fočanska 35, kod kojeg te emulatora možete i kupiti.

AT-speed i AT-once treba ugraditi slično kao i PC-speed, dakle prilimiti na MC68000.

Konstrukcija i delovanje

Oba AT emulatora koncipirana su po uzoru na PC-speed. Na pločici,

koju neposredno povežemo s motorolom u ST-u, u SMD tehnici je postavljeno relativno jednostavno kolo s Intelovim procesorom 80C286. Novi procesor preuzima atarijevih 10 MHz i nešto malo električne energije koja mu treba za rad. AT-speed npr. troši 0.5 A struje, od ukupno 3 A, koje daje napajanje ugrađeno u ST.

Ugrađeni emulator ne ometa pravilan rad programa na atariju ST. Posle ugradnje, računar se ponaša jednako kao i pre. Ali zato AT-speed nudi zanimljivu mogućnost: iz MS-DOS-a možete probuditi «uspavani» motorolu, dodeliti joj posao (kod kojeg može sudelovati i sva periferija) i nakon toga se vratiti intetu. U stvari MC68000 ba ne odmaru, jer pri emulaciji delimično saraduje s 80286, samo da je za korisnika MS-DOS ta suradnja neprimetna. Kod «buđenja», u stvari, radi se o privremenom zaustavljanju Intela, koji s lakoćom pričekava da se zadatak u MC68000 završi. Registre motore treba pre upotrebe spremili, a nakon upotrebe vratiti na staro mesto.

Test

Emulatore smo isprobali u sledećoj konfiguraciji:

- računar ATARI MEGA ST4 (4 Mb)
- tvrdi disk MEGAFILE (85 Mb/ms)
- štampač NEC P6plus
- vanjska 5,25-inčna disketna jedinica (40/80 tegova).

Na prvi pogled (izrada štampanog kola, kvalitet dokumentacije), Vortexov AT-once izgleda dosta bolja od konkurenta. Pločica je veoma lepa i precizno napravljena, čak je i na-

PROCESSOR SPEED BENCHMARK TESTS				
SUMMARY OF RESULTS				
TEST NAME	TIME IN SECONDS	SPEED INDEX		
		A vs. B	A vs. C	
INSTRUCTION MIX	12.85	0.7	2.5	
128K NOP LOOP	4.34	1.0	2.3	
DO-NOTHING LOOP	8.06	0.7	2.0	
INTEGER ADD LOOP	3.79	0.6	2.6	
INTEGER MULT LOOP	2.58	0.9	3.9	
STRING SORT & MOVE	4.28	0.7	2.5	
PRIME NUMBER TESTS	6.26	0.7	3.8	

Machine A = This V30
Machine B = 8 MHz IBM-AT
Machine C = 4.77 Mhz IBM-PC

Any Key Resume Testing Ren Exit

COPYRIGHT (C) CHIPS AND TECHNOLOGIES INC., 1986

09/07/90		BENCHMARK PROGRAM - VERSION 1.50			13:53:29	
BENCHMARK PERFORMANCE RELATIVE TO =>		IBM/PC	IBM/AT	COMPAQ	ACTUAL	
		4.7Mhz	8Mhz	3M6	MIPS	
GENERAL INSTRUCTIONS	1.93	0.56	0.28	0.32		
INTEGER INSTRUCTIONS	3.87	0.60	0.37	0.65		
MEMORY TO MEMORY	1.84	0.57	0.32	0.43		
REGISTER TO REGISTER	5.08	0.66	0.38	0.91		
REGISTER TO MEMORY	1.89	0.57	0.31	0.58		
OVERALL PERFORMANCE	2.45	0.59	0.29	0.58		



PROCESSOR SPEED BENCHMARK TESTS

SUMMARY OF RESULTS

TEST NAME	TIME IN SECONDS	SPEED INDEX	
		A vs. B	A vs. C
INSTRUCTION MIX	13.55	0.8	2.6
128K NOP LOOP	4.38	1.0	2.4
DO-NOTHING LOOP	4.05	0.9	2.4
INTEGER ADD LOOP	2.60	0.8	3.5
INTEGER MULT LOOP	1.41	0.9	7.3
STRING SORT & MOVE	3.73	0.6	2.9
PATH NUMBER SIZE	5.27	0.6	2.9

Machine A - This 7.1 Mhz 80286
Machine B - 8 Mhz IBM-AT
Machine C - 8.77 Mhz IBM-PC

Any Key Resume Testing Esc Exit

COPYRIGHT (C) CHIPS AND TECHNOLOGIES INC., 1986

10/22/91	BENCHMARK PROGRAM - VERSION 1.20			14:27:51
BENCHMARK PERFORMANCE RELATIVE TO =>	IBM/PC 4.7Mhz	IBM/AT 8Mhz	COMPAQ 386	ACTUAL MIPS
GENERAL INSTRUCTIONS	3.65	0.77	0.39	0.44
INTEGER INSTRUCTIONS	5.68	0.89	0.39	0.95
MEMORY TO MEMORY	2.53	0.76	0.44	0.60
REGISTER TO REGISTER	6.68	0.87	0.36	1.20
REGISTER TO MEMORY	2.67	0.80	0.44	0.82
OVERALL PERFORMANCE	3.40	0.82	0.40	0.80

BT-System Information: Advanced Edition 4.50. (C) Copy 1987-88, Peter Norton

Copyright Notice: (C) PC-SPEED BMB V2.11 BY JACK ELECTRONIC BMBH 1989
Operating System: DOS 3.30
Main Processor: NEC V30
Co-Processor: None
Serial Port(s): 1
Parallel Port(s): 1
Video Display Adapter: Color/Graphics (CGA)
Current Video Mode: Text, 80 x 25 Color
Available Disk Drives: B: A: - B:

DOS reports 704 K bytes of memory:
140 K-bytes used by DOS and resident programs
564 K-bytes available for application programs
A search for active memory finds:
840 K-bytes main memory (at hex 0000-0A000)
128 K-bytes display memory (at hex 0A000-0C000)
128 K-bytes extra memory (at hex 0C000-0E000)
3 000 K-bytes extended memory (at hex 10000-3F000)

Computing Index (CI), relative to IBM/AT: 4.3
Disk Index (DI), relative to IBM/AT: Not computed. No drive specified.

Performance Index (PI), relative to IBM/AT: Not computed.

PC-speed

što manja od one kod AT-speeda. Dokumentacija je napisana na nemačkom i engleskom jeziku s lepo nacrtanim crtežima za pomoć pri ugradnji. Kod PC-speeda, uputstvo je samo na nemačkom jeziku, štampa je lošija, a umesto crteža ovde su nekvalitetni crno-beli animci.

Utisci se sasvim promene kad započnemo s isprobavanjem pouzdanosti rada. Ovdje AT-speed ima veliku jednu prednost: nijedan prekid u delovanju sistema u mesec dana testiranja. AT-once mnogo problema prčinjava već i sam start MS-DOS sistema. Kada start konačno uspe, bar jedan »raspad« sistema na sat uliva dodatno nepoverenje. Zašto takva razlika u delovanju? Firma Sack je već sakupila mnogo iskustava sa svojim prvim emulatorom, a Vortex je na tom području tek početnik. Možemo se samo nadati da za poteškoće nije kriv hardver AT-oncea, nego samo prateći softver. Tako bar pokazuju iskustva s PC-speedom, gde se hardver uopšte nije menjao, s početne poteškoće su s novijim verzijama softvera (BIOS) nestajale. Kod AT-oncea možda smeta još i to da emulator nije zasnovan tako jednostavno kao AT-speed (Intelov procesor 80286 i nešto jednostavne logike), nego su

spomenutom sastavu dodali još procesor za posebne zadatke, napravljen izričito po nacrtu naručioaca (engl. custom design). Takav čip ne spada među standardne modele procesora i obično na odgovarajućem široj upotrebi. Greške se možda skrivaju i ovde.

Kompatibilnost

Kompatibilnost je kod AT-speeda nešto veća nego kod konkurenta. Ali ipak ne važi da sve što radi s AT-oncaom radi i s AT-speedom. Kod AT-oncea naročito smeta da sistem zabiokira ako neki program ne radi, dok je to kod AT-speeda retkost.

Na oba emulatora isprobali smo nekoliko najraširenijih programa: Lotus 1-2-3, Turbo Pascal 5.0, Turbo C 2.0, Gilpper, dBase III Plus, PKARC, PKZIP, PC Tools, Norton Utilities, Tetris... Svi su radili kako treba. Poteškoće su nastale s AT-oncaom, koji ni hteo da učila sistem MS-DOS 3.3 iz particije C. AT-speed je to napravio s lakoćom. Zahtev za pravilan rad bio je: ponovo formatiraj particiju C. Kako je na particiji C bilo oko 32 Mb podataka i programa, zahtev je bio malo preostar, pa zato s Vortexovim emulatorom nismo isprobali kako se sistem učitava.

U memoriji iznad 840 K oba se emulatora jednako dobro snalaze. Sistem za sebe prisvoji 704 K, a kako smo na raspolaganju imali 4 Mb RAM-a, emulatori su ga dobro koristili kao EMS (proširenu) i XMA (produženu memoriju). Pri tome je AT-speed bio nešto sprtniji, jer čak zna i da deli memoriju s atarijem ST: program za pokretanje emulatora može biti instaliran kao pomoćna datoteka (accessory - .ACC). Tako je MS-DOS pristupačan iz bilo kojeg programa u računaru, pod uslovom da program ostavi dovoljno prostora u memoriji. Pri skoku u MS-DOS, računar zapamti šta je radio pre toga, dok je još bio atari, a kod povratka nastavi tamo gde je njegov rad bio prekinut.

Obzirom na emulatora od pre godinu dana, oba su brža za oko 50%. Za upoređenje: kod PC-speeda, Nortonov faktor je bio 4.2, a kod AT-speeda 6.7. AT-onca je tek nešto sporiji (vidi upoređenje brzine između XT i AT emulatora). Lepo se vidi razlika u brzini između NEC- V30 i Intela 80286 pri istom kloku (8 MHz). Detaljnije testiranje (MIPS) je pokazalo da AT-speed postiže 85-procentnu brzinu AT kompatibilna na 8 MHz. Razlika u brzini oba AT emulatora je skoro zanemariva.

Posebnosti

Već smo spomenuli sprtno korišćenje memorije iznad 704 K. AT emulatori ne mogu opravdati svoje ime ako ne bi mogli raditi s nekim programima u zaštićenom načinu (protected mode). Po izjavi proizvođača, na oba emulatora mogu raditi Windows 286 3.0. Za to je potreban atari ST s više od 1 Mb memorije. Ali, AT-onca zna nešto što AT-speed nije sposoban: BLINKing. Izgleda da to leži u onom posebno napravljenom čipu. Svejedno, ta karakteristika nije dovoljna da bi AT-onca nadmašio AT-speed. Spomenimo još da se u radu s AT-speedom fantovi mogu izmenjivati s onima s TOS-u.

Koji je emulator bolji? Besumnije AT-speed. Svi su argumenti za takvu tvrdnju sažeti u tri reči: pouzdanost, kompatibilnost i brzina. Raduju činjenica da borba još nije završena Vortex će verovatno uskoro imati dovoljno iskustva i može se dogoditi da će zbog svojeg zaista profesionalnog (i) pristupa i pažljivog odnosa prema kupcu i uspeti.

U Ujedinjenoj Nemačkoj AT-speed košta 600, a AT-onca 450 DEM (takve cene preporučuju proizvođači).

AT-speed

BT-System Information: Advanced Edition 4.50. (C) Copy 1987-88, Peter Norton

Copyright Notice: (C) AT-SPEED BMB V2.11 BY JACK ELECTRONIC BMBH 1989
Operating System: DOS 3.30
Main Processor: Intel 80286
Co-Processor: None
Serial Port(s): 1
Parallel Port(s): 1
Video Display Adapter: Color/Graphics (CGA)
Current Video Mode: Text, 80 x 25 Color
Available Disk Drives: B: A: B:

DOS reports 704 K bytes of memory:
92 K-bytes used by DOS and resident programs
612 K-bytes available for application programs
A search for active memory finds:
630 K-bytes main memory (at hex 00000-0A000)
128 K-bytes display memory (at hex 0A000-0C000)
64 K-bytes extra memory (at hex 0C000-0E000)
2 233 K-bytes extended memory (at hex 10000-3F000)

Computing Index (CI), relative to IBM/AT: 6.7
Disk Index (DI), relative to IBM/AT: Not computed. No drive specified.

Performance Index (PI), relative to IBM/AT: Not computed.

PROCESSOR SPEED BENCHMARK TESTS

SUMMARY OF RESULTS

TEST NAME	TIME IN SECONDS	SPEED INDEX	
		A vs. B	A vs. C
INSTRUCTION MIX	11.42	0.8	2.6
128K NOP LOOP	4.28	1.0	2.4
DO-NOTHING LOOP	4.01	0.9	2.5
INTEGER ADD LOOP	2.76	0.8	3.5
INTEGER MULT LOOP	1.34	0.9	7.3
STRING SORT & MOVE	3.10	0.8	2.9
PATH NUMBER SIZE	5.22	0.8	2.9

Machine A - This 8.2 Mhz 80286
Machine B - 8 Mhz IBM AT
Machine C - 8.77 Mhz IBM-PC

Any Key Resume Testing Esc Exit

COPYRIGHT (C) CHIPS AND TECHNOLOGIES INC., 1986

10/02/90	BENCHMARK PROGRAM - VERSION 1.20			09:44:43
BENCHMARK PERFORMANCE RELATIVE TO =>	IBM/PC 4.7Mhz	IBM/AT 8Mhz	COMPAQ 386	ACTUAL MIPS
GENERAL INSTRUCTIONS	2.66	0.77	0.39	0.44
INTEGER INSTRUCTIONS	5.75	0.90	0.39	0.96
MEMORY TO MEMORY	2.56	0.79	0.44	0.61
REGISTER TO REGISTER	6.75	0.88	0.37	1.21
REGISTER TO MEMORY	2.70	0.81	0.45	0.83
OVERALL PERFORMANCE	3.43	0.83	0.41	0.81

Lisica sa pedigreom

DAVOR PETRIĆ

Baze podataka su jedno od neopodnih oruđa poslovnog svijeta. Iako statistike pokazuju da je najveća primjena kompjutera za obradu teksta vrlo blizu i u korištenje za obradivanje baza podataka.

Tu je stanje godinama bilo mirno. Dominirao je dBase III+, a neki su se patili sa Clipperom. Došao je i dBase IV, ali u našim krajevima, a niti u bijeloj svjetlosti, nije bio baš sjajno prihvaćen. Jedna od najvećih zamjerki dBase programima je da su spori. Sa druge strane, Clipper daje veću brzinu, ali nije baš ugodan za rad.

Tako se unazad nekoliko godina sve češće počela spominjati FoxBase baza podataka. Polako je gradila svoju popularnost, a u prošloj godini, na tržištu je izbacila i jako poboljšanje, FoxPro. Mi smo se potrudili da paket dobijemo na test, pa vam nudimo naše iskustvo.

Verzija koja nam je isporučena na testiranje je FoxPro Lan 1.01, i datirana je 23. 8. 1990. godine.

Kako izgleda

Kutija je dosta teška, kao što priliči programu za bazu podataka. Kada je otvorile u njoj nalazite set disketa i literaturu. Diskete su isporučene i u formatu od 3.5" (3 kom) i u formatu 5.25" 360 K (5 kom). Po tome nalazite još i dvije demonstracione diskete za grafički program koji koristi podatke iz FoxPro.

Literature ima dosta, i solidno je napisana. Iako su knjige u mekani uvezu, moguće ih je ostaviti otvorene na stolu, bez da se unštavaju ili da se same zatvaraju. Na 30 stranica, objašnjene su razlike između verzije 1.01 i prve verzije, sa brojem 1.00. U drugoj knjizi, od 580 stranica, obrađeni su detalji vezani uz rad na mreži. Za rad na mreži potrebno je 480 K slobodno memorije na radnim stanicama (nakon što je učitana network shell). Podržani su svi glavni sistemi mreže.

Slijedi Tutorial, knjiga sa nekih 200 stranica. Postepeno vas uvodi u mogućnosti programa, naravno vježbanjem na priloženim primjerima. Vježbe obuhvaćaju dobar dio interaktivnog rada.

Druga knjiga je User's Guide, i sadrži objašnjenja potrebna za interaktivni način rada, opisujući sve menije i sve moguće izbore u njima. Sadrži negdje oko 480 stranica. Između ostalog, ovdje su navedene i greške koje možete dobiti. Detalji oko prilagođavanja programa vašim zahtjevima, dodacima programu, i naprednijim temama, također su tu.

Posljednja, a ujedno i najdeblja knjiga bit će najviše korištena, jer je ona referentni vodič svih funkcija i komandi koje ćete koristiti pri pro-

gramiranju FoxPro baza podataka. Sadrži oko 660 stranica.

Posljednja je ostala knjižica, koja na 48 stranica navodi sve funkcije i komande sa sintaksom i objašnjenjem od jedne rečenice. Očito to je brzi referentni vodič.

Ovdje moramo reći da je u paketu trebala biti još jedna knjižica, ona koja dokumentira module FoxView, FoxCode i FoxDoc. Na žalost, nije nam stigla u paketu (ali smo zato dobili dva referentna vodiča), a niti na naše zapamaganje, služba za podršku nije reagirala.

To nas vodi žalosnom zaključku da Fox Software kompanija spada u one koje baš ne mare za korisnike njihovih programa, nakon što im jednom prodaju svoj program. Takvu odluku zaslužuje i firma WordPerfect, dok je Quarterdeck dosta usporan, ali ipak radi. Za razliku od njih, možemo pohvaliti npr. Bor-



land, Corel, Central Point Software, Mark Williams Company (Coherent) i Microsoft. Napominjemo da se sve ove primjedbe odnose na naša lična i iskustva naših poznanika. Možda vi imate drukčija iskustva.

U korisničkom vodiču imamo master indeks, koji integrira indekse u svim knjigama. Ideja je dobra, mada se indeksi svode na detaljnije tabele sadržaja, sa navedenim samo ključnim riječima. Ne može ipak svatko imati uputstva kvalitete Borlanda, Microsofta i Mark Williams Company.

Instalacija u vesela. Ubacite disketu u kompjutor, otkucajte install i - kaže da nema mjesta. Morate malo pogledati u uputstvo. Nakon toga, sve ide lijepo. Ovisno o vašim željama, ne morate instalirati cijeli program, već samo neke module. Bit će vam prikazano što možete birati i koliko prostora na disku treba svakom dijelova. Instalirani program zauzet je nam oko 4 Mb diska.

Nastojte ne izgubiti jednu ceduljicu sa hrom brojeva koja vam je došla sa paketom. Tamo se nalaze

šifre bez kojih možete sjesti i plakati. Kod svake reinstalacije treba ih ponovo kucati, zato oprez.

Novosti

FoxPro u razvojni sistem sa vrlo velikom mogućnostima i velikom lakoćom rada, posebno ako ga uspoređujemo sa dBase programima. Između ostalog, i zbog rada u prozorima, izuzetno je olakšan razvoj aplikacija jer je moguće u trenutku otici bilo gdje (i u samoj aplikaciji, ali i u bazama i svim ostalim prozorima), pogledati relevantne podatke i vratiti se, pa čak i integrirati ih izvođač ih interaktivno i prebacivši ih u aplikaciju.

U verziji 1.01, u odnosu na verziju 1.0, osim čišćenja programa i osuvremenjavanja literature (sve je u knjigama, ne treba vam README datoteka), dodani su detalji poput mogućnosti specificiranja konfiguracijske datoteke pri dizanju programa, zabranjivanje upotrebe expanded memorije, moguće je precizirati lokacije datoteka, neki detalji oko interfacasa su poboljšani, i, naravno, dodane su neke nove komande.

FoxPro je potpuno kompatibilan sa FoxBase + i dBase III, a sa dBase IV u kompatibilan na nivou jezika. U radu je osjetno brži, koliko ovisno je o aplikacijama, ali faktor je uvijek lako uočljiv. To je ujedno jedna od glavnih prednosti ovog programa nad dBase. Mnogi korisnici koje smo čuli, govore da će preći (ili su već) prešli sa dBase na FoxPro upravo zbog veće brzine rada.

Jedno u čemu nije kompatibilan su master indeks format (.MDX) datoteke generirane iz dBase IV, ali ne zato što je FoxPro loš, već zato što je jednostavno nadokup dBase. Indeks i samo polja nisu kompatibilni. Ali zato bolji!

Što se tiče kvalitete programskog jezika, dBase je također potučen. Oko 160 poboljšanja jezika je ugrađeno u FoxPro u odnosu na dBase IV. Jedino što je za neke korisnike nezgodno, jest što program ne generira samostalni izvršni kod (sa EXE nastavkom), već je za izvršavanje programa potreban ili FoxPro ili njegova Run Time verzija. Olakšavajuća okolnost je što jednom kupljen Run Time paket omogućava neograničen broj instalacija aplikacija, a bez ikakvih problema sa autor-skim pravima.

Interesantno je da će se program sasvim solidno izvršavati i na XT sistemima sa samo 512 Kb RAM-a. Probajte u tim uvjetima potjerali dBase IV. Istina da nećete biti jako sretni niti sa FoxPro paketom u takvoj okolnosti, ali radili bi sasvim izdržljivo.

Iako u tako malih osnovnih zahtjeva, sve što mu date iskoristit će vrlo rado i vrlo dobro. EMS 4.0 memorija i matematički koprocesor će

ga osjetno ubrzati. Po svemu sudeći, a za razliku od dBase IV, FoxPro je vrlo korektno napisan program bez većih bagova i problema. Jedan smo ipak pronašli, mada nije isključeno da se pravo objašnjenje krije u knjizi koju nemamo. Kada imate postavljen radni direktorij različit od onoga u kome držite programsku datoteku FoxPro programa, i pokušate startati FoxView, FoxCode ili FoxDoc, FoxPro ih ne može naći i cvili da ne postoje. Naprosto ne možemo vjerovati da mu se ne može objasniti da ti izvršni programi nemaju što raditi u direktoriju sa datotekama. Kada promijenite direktorij, sve radi kako treba.

Jedna od ne malih prednosti Foxa je što postoje njegove verzije i na Macintosh kompjutorima kao i za Unix, tako da će vaše aplikacije i podaci bez ikakvih promjena moći da se izvršavaju i na tim sistemima. Uostalom, na Mekovima je Fox praktički standard za baze podataka.

Pošto je očito da u programima te vrste postoje dva načina rada, interaktivni i onaj oko pisanja vlastitih aplikacija, tako ćemo nastojati i opisivati ovaj paket.

Interaktivni rad

Korisnički interfacas u visokom nivou, i zalista je pratećio jako puno prednosti grafičkih korisničkih interfacasa u tekstualni režim rada. Kažu da je to jedna od najuspješnijih kopija Mekovog načina rada, a to znači da je stvar dobra. Po nama, od tekstualnih interfacasa sa prozorima, jedini koji je bolje urađen jest onaj QuattroPro.

Nežnost u takvim okolnostima jest i miš. Da bi vam olakšali posao mislim, uz program su priloženi i upravljački programi (device drivers) za Microsoft i Logitech miša, mada je program tolerantan što se tiče verzije drajvera kojim raspolazete.

Ako nemate miša, sve se može uraditi, ali kao i sa ostalim aplikacijama orijentiranim mišu, npr. Windows aplikacijama, rad bez miša je puno sporiji i komplikiraniji. Danas ipak miš ne košta puno, kupite ga. Imajte na umu da sve više današnjih aplikacija, u stvari skoro sve koje ista vrijede, rade sa mišem.

Mi smo vidjeli i radili sa mnogo miševa i surugata, i jedan miš je odskočio iz te hrpe toliko da dugo iza njega nema nikoga. Radi se naravno o pravom Microsoft mišu. Njegov dizajn i kvaliteta izrade su apsolutno bez promca. Nemojte se bi nepotrebno komplicirati život kupovinom nekakvog drugog miša. Čim ga jednom probate, shvatite da ste do tada živjeli u mraku.

Da se vratimo na FoxPro. U prvom redu ekrana se nalazi meni, naravno sa padajućim (drop down) podmenijima iz kojih birate komande. Izbor komandi odlično radi sa mišem, i moguće je vrlo jednostav-

no mijenjati diskove, direktorije ili tipove datoteka (programe, pogleda, izvještaje, baza...) koje želite vidjeti izlistane.

Sve se odvija u prozorima. Prozore možete relocirati i mijenjati im veličine. Način gledanja na datoteke je zaista dobar. Vrlo je zgodna mogućnost dijeljenja prozora sa datotekom tako da reorganizirate pogled koji imate na datoteku. Možete npr. zamrznuti prvi stupac (u browse načinu rada) na ekranu, a nekoliko alijedećih preskočiti i dovući na ekran npr. stupce 4 do 6. Opet i njima možete prilagoditi širinu tako da vam odgovaraju «olščina» organizacija podataka koje vidite.

Koliko vam to znači jasno je svima koji imaju npr. ime ili prezime u prvom polju, a podatke koje trebaju gledati, negdje u 8. polju. Uvijek je borba sa time kako vidjeti istovremeno (obično nikako) i ime mušterije i podatke koji vam trebaju.

Da bi stvar bila zgodnija, moguće je istovremeno podatke gledati i u browse i u edit načinu. Osim tih prozora možete otvarati velik broj ostalih (više nego što ćete ih imati živaca gledati i boriti se sa njima). Pri svemu tome uživajte u brzini rada sa tekstualnim načinom rada vašeg «omputera».

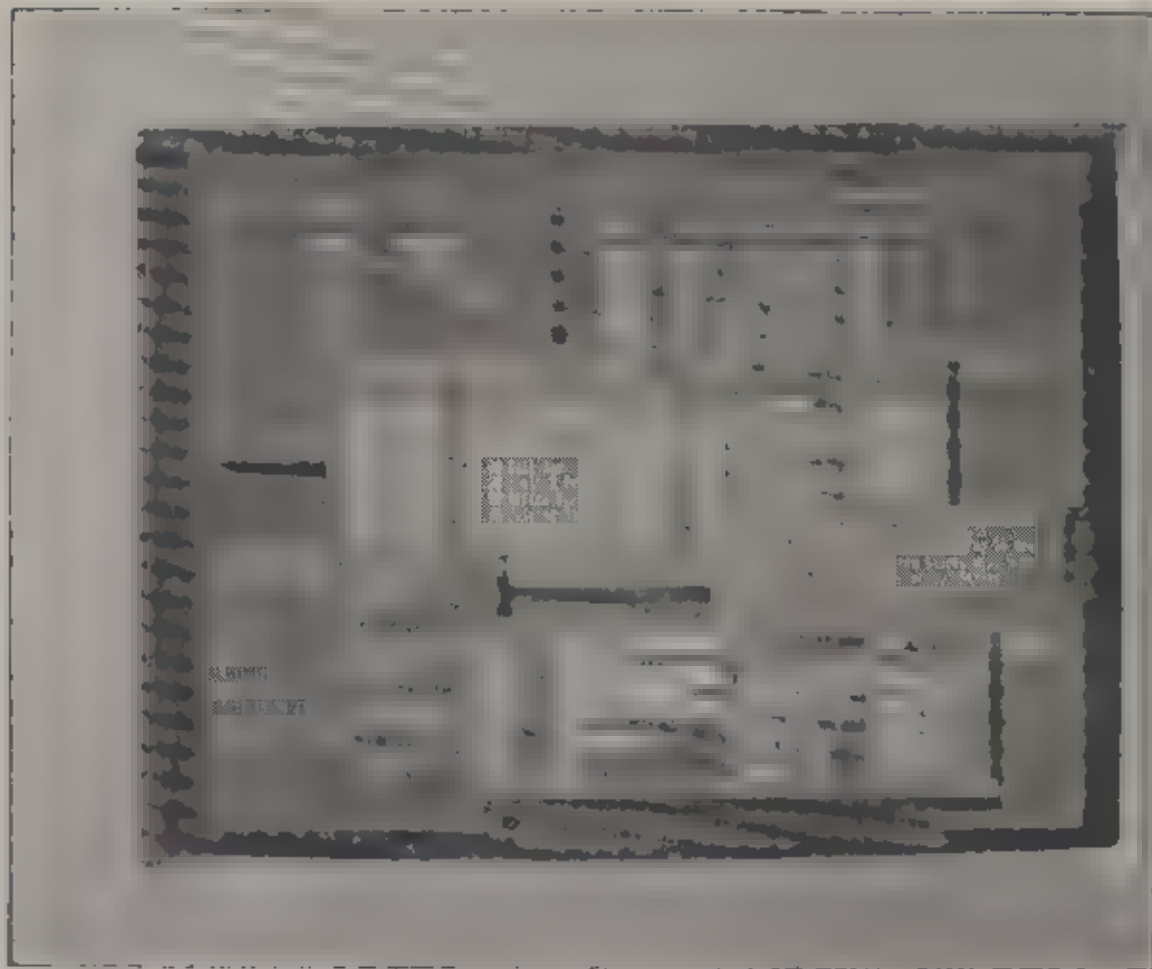
Prilikom postavljanja uvjeta za npr. kopiranje pojma datoteka u drugu datoteku, sve je uvjete moguće postaviti mišem. Naravno, tekst upita morat ćete ipak upisati. U View prozoru se postavljaju i radne površine kojima želite raditi i određuje se koju će datoteku učitati; odja istovremeno moguće je imati otvorenih do 25 baza podataka. Tu ćete postavljati i uklanjati veze među bazama (što se tiče indeksa u jednom trenutku, može biti otvoreno do 25 indeksa za svaku bazu podataka).

Važno je spomenuti i da se tu krije tzv. On/Off panel, kojim uključujete i isključujete različite opcije konfiguracije programa (da li pita prije prepisivanja datoteke, da li zvoni, da li je uključen razvoj programa, pomoć kako tretirati staze (ostali slični elementi). Ukupno ih je 26.

Jedan od prozora koji je najmiliji stariim ykovima (ubiteljima dot prompta je naravno, komandni prozor) njega možete otvoriti i zatvoriti, a u njemu se pojavljuje tekst svake naredbe koju izdate mišem, odnosno ako vam je tako draže tamo možete upisivati komande direktno.

Jedna od primjena ovog prozora jest pri programiranju. Pošto je u nekom komplikiranom uvjetu vrlo lako izostaviti neki detalj, najzgodnije je konkretan uvjet postaviti mišem u interaktivnom načinu rada, a onda cijelu stvar iz komandnog prozora prebaciti u prozor u kome vam se nalazi program. Prozori sa tekstom također mogu biti brojni, a jednostavno je seljenje podataka među njima. Veličina programa koja možete upisati u te prozore nije ograničena slobodnom memorijom već samim slobodnim prostorom na disku i, naravno, dozvoljenom broju otvorenih datoteka (sa Files u CON-FIG.SYS).

Editiranje je solidno, postoje



zamjena i traženje teksta (vrlo brzo) i cut/copy/paste mogućnosti i naravno undo/redo. Preference možete postavljati i snimati ih na disk kako biste editor prilagodili svojim zahtjevima.

Programiranje

Već smo rekli da je broj instrukcija vrlo dobar. Kao da to nije dosta, omogućeno je i efikasno dodavanje funkcija koje korisnik definiira sam. Usprkos vrlo dobroj mogućnosti definiranja korisničkih funkcija (User Defined Functions), tu bi se moglo još poneti detalj i unaprijediti.

U helpu su navedene sve funkcije i komande tako da je u tome elementu pomoć sasvim dobra. Za svaku naredbu ponuđeni su primjeri, a sve je sasvim dobro objašnjeno i u referentnom priručniku za funkcije i komande. Dodatna pomoć, jednostavan ali efikasan štos, jest navođenje funkcija i komandi po kategoriji posla koji obavljaju. Abecednim redom su navedene funkcije iste grupe (npr.: one koje rade sa znakovima, brojevima, datumima i vremenom, logičke za manipulaciju varijablama stringovima, bazama, slogovima datotekama, indeksima, relacijama, izvršavanjem programa štampanjem, labelama prozorima ili tastaturom da spomenemo neke od grupa). Vodič je koristan jer će na jednom mjestu biti sve što vam može pomoći da smislite kako da što bolje riješite neki konkretan problemčić a svi znamo da se ne

možemo baš uvijek sjetiti što nam taj trenutak treba. Jest nam na vrhu jezika ali... Ne zaboravite da se ovdje radi o nekakvih 500 navoda koje treba imati stalno u glavi.

Jedna od stvari koja je riješena kako treba i zaslužuje pohvalu jest mogućnost ustanovljavanja praktično bilo koju kombinaciju tastera da je pritisnuta, i, naravno, njeno korištenje u daljnjem dijelu programa. Set instrukcija koji rade sa tasterima je vrlo bogat, a isto tako i mogućnost kontroliranja reakcija na taster (ili kombinaciju tastera).

Prilikom razvoja aplikacije konstrukcija interfeasa prema korisniku bit će vrlo jednostavna, pošto Fox-Pro omogućava jednostavnu upotrebu prozora konstrukcije menija. Čak će i početniku to biti lako koristiti.

Odlučna je mogućnost rada memo poljima: stringovima. Memo polja mogu biti bilo koje veličine mogabajti dva (ili više) noma problema. Stringovi memorijskih varijabli mogu biti do 84 K dužina. Snalaženje u tako velikim memo poljima je odlično jer postoji velike mogućnosti traženja. Osim teksta, memo polje može sadržavati i grafiku, digitalizirani zvuk i npr. EXE datoteke, mada trenutno te mogućnosti još nisu jako dobro riješene u praksi.

Prevodilac (takav kakav jest) je integriran u sam glavni program. Greške pri prevodenju može locirati i u datoteku a jedino ograničenje jest da mu je maksimalna dužina programske linije 1024 bajta.

U razvoju aplikacija pomagal će

la se ugrađenim debuggerom. U njega unesite koje vas varijable zanimaju, program potjerajte u tražeći prozoru i imate sve relevantne podatke pred očima. Da bi postavili prekidne točke dovoljno je na željenu poziciju kliknuti mišem a kroz program se možete kretati i pojedinačnim koracima.

Osim spomenute mogućnosti direktnog generiranja izvršanih datoteka (koje se izvršavaju bez kupovine Run Time modula), nedostaci su i nemogućnost direktnog linkovanja modula napisanih u C ili assembleru. To je danas zahtjev koji svaka baza podataka mora pružiti, jer su baze uvijek velike i kao da se nikada ne izvršavaju dovoljno brzo. Assembler-ski, ili onaj napisan u jeziku C, modul postavljen na pravo mjesto može zapanjujuće puno ubrzati određene tipove aplikacija. Istina je da mnoge aplikacije rade vrlo dobro i bez ove mogućnosti a da ona često ostaje neupotrijebljena jer se ljudi tek privikavaju na takve mogućnosti.

Također u ovim današnjim vremenima lako bi vam mogle zatrebati i SQL mogućnosti u vašoj bazi podataka, a to je obećano u slijedećem izdanju ovog paketa. Istina je volju, danas je jako teško pronaći optimalno rješenje za SQL na našim malim sistemima.

Dodatni programi

Iako baš i nisu dodatni programi u pravom smislu riječi spominjemo

ih ovdje. Radi se o opciji System iz glavnog menija. U njoj je i komanda za rad sa datotekama (znate ono premještanje kopiranje i ostale stvari). Za jednostavne poslove, poslužit će sasvim dobro. Ipak, malo kompleksnija radnja komplicirat će stvar i poželiti ćete otići u DOS i upaliti PC Tools V6.

Tamo se nalazi i poziv u pomoć, ali radi se samo o indeksnoj pomoći. Ništa od kontekstno senzitivne pomoći kada zaglavite u interaktivnom načinu rada. Tu ćete pronaći i neke sitnice kao što su kalkulator, kalendar i rokovnik, te jednu malu igru (puzzle). Najvjerojatnije je da će vaš PC Tools V6 Desktop ostati u memoriji.

Rekli smo da uz FoxPro dolaze i neki dodatni programi, ali njih, na žalost, možemo samo načelno opisati, sjećate se da nam fall jedna knjiga.

FoxView je program za dizajniranje ekrana (Screen Designer), i trebao bi vam omogućiti da radite vrlo jednostavno. Omogućava istovremeni pogled na podatke više baza i uključivanje cijelih programa u ekran koji se uređuje. Ekranu mogu biti kreirani i tako da koriste polja iz više odvojenih baza podataka, što zna biti jako korisno.

FoxCode je generator aplikacija. Generira sve što vam treba (tako kažu). U njega je uključen jezik za kreiranje maski (Templates) koje opisuju strukturu programa koji želite generirati. Neke od maski su vam isporučene s programom, a svoje maske možete generirati u toj samostalnoj i kompletnoj radnoj okolini.

FoxDoc je vrlo zgodna stvar i njega ćete koristiti da biste dokumentirali svoje programe. Jasno vam je da nedokumentiran program, nakon kratkog vremena postaje prava enigma kada je potrebno promijeniti neku sitnicu. Zato će FoxDoc mapirati tok (jednog programa ili cijelog sistema svejedno) i napraviti kompletnu specifikaciju, skoro bez vašega truda.

Podaci će uključivati gdje su koja imena varijabli spominjana, kre-

irani ili mijenjana, dijagram strukture odnosa svih modula koji čine vašu aplikaciju, uniformnu kapitalizaciju ključnih riječi, indekse i na kraju jasna zaglavlja modula. Naravno, to je samo neke od važnijih stvari. Ovo je vrlo koristan alat za one koji rade na razvoju aplikacija.

Uz program dobijate samo demo verziju FoxGraph aplikacije. Taj program omogućava da podatke iz svoje baze pretvorite u veliki broj (ponekad i čudnih oblika) grafikon. Naravno, radi se o 3-D perspektivi, a mogućnosti kontrole izgleda i tipa grafa olakšavaju izradu preglednog 3-D grafikona. Pri tome je potrebno samo nekoliko klikova da se odredi tip grafa, a ostalo se radi automatski. Nama demo izgleda više kao igra, jer ne vjerujemo da liko može pronaći svrhu za neka od pogleda na graf koji su tamo prikazani.

Prednosti i mane

Jasno da će svaki dBase kompatibilan program tvrditi da je bolji od uzora, da je brži, da ima bolje indekse, a što je od svega toga istina, često ostaje krajnjim korisnicima da se uvjere.

Što se tiče programiranja, mogućnosti jezika i interaktivnog rada, FoxPro je u velikoj prednosti nad dBase programom.

Drugo važno pitanje je brzina. Što se tiče brzine i kompaktnosti indeksa, FoxPro je bolji od dBase IV. Tipično poboljšanje u brzini je oko 50%, što znači da dobijate u velikom indeksu nije tako spektakularan. Indeks će biti kraći i do 15, ili čak 20-tak postotaka ali u nekim kompleksnijim slučajevima, mogu biti ponešto i duži od onih koje generira dBase IV ali ne bitno. Brzina ipak ostaje uvijek.

Sve u svemu, sasvim dobar alat za profesionalce koji razvijaju aplikacije i od toga žive, ali i za nekog tko radi sa puno podataka, a nije ni voljan niti spreman učiti programiranje.

U nekim aspektima (povezivanje sa C i Assemblerom, direktno provođenje izvršnog .EXE koda, SQL povezivanje) bolji su Paradox i Clarion, ali sigurno je da će FoxPro (još ako poboljšaju odnos prema korisnicima) zadovoljiti velik broj korisnika koji žele direktno raditi sa sporog dBase na nešto puno bolje a bez dodatnog učenja ili mrdanja sa prilagodbom postojećih aplikacija.

Jedina dijena koju znamo točno jest ona za običan FoxPro, ostale cijene primite sa rezervom, i samo kao grubu orijentaciju.

Cijene u USD
FoxPro 1.01 - 795
FoxPro LAN - oko 995
Run Time Unlimited Pack - oko 495.

Adresa: Fox Software, Intl. Intech House, Cam Centre Wilbury Way, Hitchin Herts, SG4 0AP, United Kingdom. Tel.: 9943 462 421 999. Fax.: 9943 462 421 316.

Mađžari čitaju brže

BORIS HORVAT

Konačno (vidi tekst u okviru) smo dobili obećani i dugo očekivanu verziju programa za optičku identifikaciju teksta RECOGNITA, proizvod koncerna SzKI iz Budimpešte. To je program iz više cenovne klase (oko 3.000 DEM) kojim proizvođač RECOGNITA Corporation (u okviru koncerna SzKI) cilja na prvo mesto u Evropi, a pri tome ima ozbiljnog konkurenta samo u američkom programu OMNIPAGE RECOGNITA Corporation (i ceo koncern SzKI) je još jedan primer kako se znanjem i profesionalnim nastupom može prodati i afirmisati u svetu. Mađari ovo na području računarstva već dugo dokazuju (od igrice za spectrum dalje), tako da nam mogu služiti za uzor.

Već je sam paket profesionalno oblikovan. Priručnik je u lepoj plastificiranoj fascikli koja je smeštena u isto tako plastificiranu robusnu kutiju. U fascikli su u stvari dva priručnika sastavljena od perforisanih listova, tako da pojedine delove možete jednostavno odbacivati. U fascikli su takođe diskete tri 5,25-palačne, odnosno dve 3,5-palačne. Datoteke na njima su delimično komprimirane i raspakuju se kod instalacije programa. Diskete su u posebnom omotu sa upozorenjem na licenčni ugovor, na koji pristajemo kad skidamo pečat omota; sve u stilu kao što smo navikli kod profesionalnih

programskih paketa iz Amerike. Sam program je takođe profesionalno zaštićen; u paketu dobijamo utičnicu koju moramo da priključimo na izlaz za štampač računara (naravno, štampač možemo nesmetano da upotrebljavamo).

I na instalaciju programa nemoćuje je imati primedbe jer jednostavnije ne može. Instalacijski program obavlja većinu posla sam (prepoznaje video karticu, utvrđuje prisustvo produžene, odnosno proširene memorije), tako da korisnik mora da odabere samo jezik za komunikaciju (jezika ima čitavo tuce) i da navede marku optičkog čitača (možemo da biramo među oko sto aparata pedesetak proizvođača). Uz to, Recognita prepoznaje standardne grafičke dodatke (npr. tipa TIF). Bogata je i paleta tekstualnih formata u koje Recognita zapisuje identifikovani tekst; tu možemo da biramo među dva luceta, od formata ASCII do formata tekst editora, za koje još uopšte nismo čuli. Vreme instalacije odlučujemo se takođe za okolinu u kojoj ćemo upotrebljavati Recognitu da li će to biti klasični MS-DOS (sa skromnim meuglednim prozorima i mišem) ili ljuzbazna okolina MS-Windows (sa svim «plastificama» koje nudi). Treba znati da u okolini Windows mogu nastati problemi sa nedostatkom memorije i da Recognita u skromnoj okolini DOS i do 50% brže.

Kod testa smo upotrebljavali ručni optički čitač GeniScan kojeg, na žalost, nema na spisku Recognite. Žalo smo svaku identifikaciju morali da obavimo u dva koraka: najpre smo čitačem i priloženim programom generisali grafičku datoteku

Tehnike prepoznavanja tekstova

Uglavnom razlikujemo dva suštinski različita postupka optičkog prepoznavanja tekstova: optički OCR (Optical Character Recognition) - prepoznavanje karakteristika slova (feature recognition) i prilagođavanje uzorka slova (pattern matching). U prvom postupku nastojimo da utvrdimo karakteristične osobine slova - veliko slovo T, na primer, čine dve crte: vertikalna i horizontalna. Taj je postupak univerzalan, jer ne zavisi od tipa slova (fontova), ali računski je dovoljno složen. Program Omnipage koji ga primenjuje, testuje poseban kopirator za personalni računar. U drugom načinu, prilagođavanju uzorka, pročitanu sliku slova program prvo normira na standardnu veličinu a zatim je upoređuje sa slikama slova poznatih od prethodnog učenja. Taj postupak koji primenjuje mađar-

ska Recognita, računski je, istina, brži, ali ograničen u setu znakova (fontova). Program mora da ih poznaje od prethodnog da ih nauči. U ovom postupku prepoznavanja prisutan je problem dvoznosti slova. Skoro nijedan program ne razlikuje slovo O od nule, odnosno broja 0. Problema ima i u parovima I i 1, kao i S i 5. Drugi problemi u prepoznavanju teksta su sledeći: pokrivena slova (primer T ili V) povezana, odnosno spojena slova (ll ili ff), prepuna obilna naslona pokrivena štampa kurziv. Naravno, problem predstavlja i grafiko ili slike u listu, za njih je najbolje da programu sarkazeno da taj deo ne uzima u obzir. Pouzdatost prepoznavanja tekstova može da se poveća aktiviranjem programa za prepoznavanje teksta sa programom za proveravanje pravopisa (Spell-checker), koji će rečnikom utvrditi nepoznata slova ili će rečnik koristiti da proveriti ona slova za koja nije siguran da ih je pravilno prepoznao. Tu mogućnost već pruža paket programa (program za

INTERSOFT

Andrićev venac 2/1

Računari
PC AT
286,386,486

Telefax
Panasonic KXF 50
15.500,00 din.

Štampači
HP LASER JET III
52.999.din.

FUJITSU DL3400
A3 / 24 pins
18.500,00 din.

011/331-374
340-408,339-104

TIF, a onda smo tu datoteku dali na obradu Recogniti

Za vreme testiranja smo proveravali sledeće podatke iz reklamnog teksta za Recognitu

- prepoznaje različite međunarodne setove znakova a posle poslednje prezentacije uključili su i jugoslovenski set;

- prepoznaje različite oblike slova, čak u istom redu

- prepoznaje znakove veličine od 6 do 24 tačke, s proizvoljnim razmakom između znakova odnosno redova;

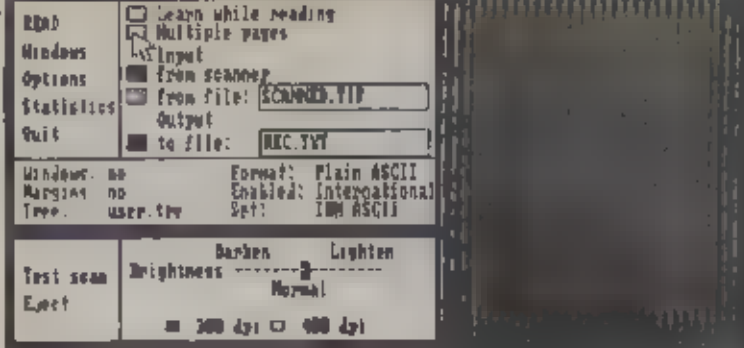
- pouzdanost identifikacije > 99.9% (manje od 1 greške na 1.000)

- interaktivno nauči da prepoznaje nepoznate znakove.

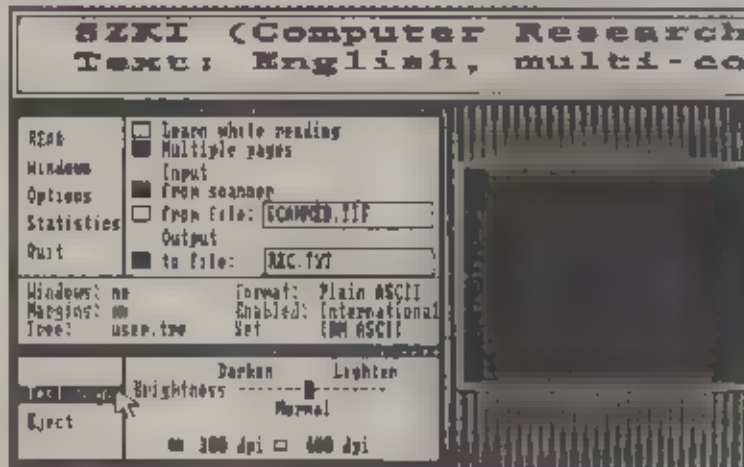
- brzina identifikacije > 200 znakova u sekundi

- može da deluje u pozadini (background) ili kao paketna obrada (batch)

Sa ručnim optičkim čitačem čoveku je potrebno nešto spretnosti odnosno vežbe, pro nego što u nesec- tekstu u računar to jest da jednako mereno i okomito na redove teksta pomera čitač po papiru. Pravilno treba podesiti i kontrast u čitaču ako je nedovoljan slova će biti «suviše suva» - loše odštampana (nezavršena) a ako je suviše snažan slova će biti «suviše debela» pa će se dodirivati - u oba slučaja program za OCR imaće probleme tako da će pouzdanost identifikacije brzo pasti ispod 90%. No, to smo brzo naučili i kreirali skoro tucе datoteka TIF iz različitih tekstualnih predložaka (knjige, revije, časopisi, matični i laserski štampači) ■ testiranje programa Recognita. Pošto nisimo još ranije pomislili koliko su velike te datoteka (sve zajedno bilo je 5 Mb prostora na tvrdom disku), u jednom trenutku smo napunili tvrdi disk. Za dalje testiranje smo kontinuirano brisali deo datoteka TIF. Samo usput smo proverili kako se



Slika 1. Glavni meni Recogniti Plus



Slika 2. Probno čitanje

ponaša program za OCR koji je bio u paketu sa čitačem. Brzo smo se razočarali jer je pouzdanost identifikacije bila između 20 i 90% - to kod istih datoteka TIF koje je kasnije Recognita s lakoćom prepoznala (to je značilo uvek preko 99%).

Recognitu smo pokretali u MS-DOS i pomirili se sa manje ljubaznom grafičkom okolinom ali zato sa nešto bržim prepoznavanjem tekstova. U glavnom meniju (slika 1) biramo između čitanja iz optičkog čitača odnosno datoteke TIF, datoteku u kojoj treba da se ispiše prepoznati tekst u kakvom obliku je tekst (uspravan engl. portrait, ili te-

žeci landscape) i da li da se program uči za vreme čitanja («Learn while reading»). Ovdje podešavamo i osjetljivost čitača (brightness) - njegovu rezoluciju (300 ili 400 tačaka na palac). Sa opcijom «Test Scan» učitavamo deo teksta iz čitača na ekran, da proverimo kvalitet slike i po potrebi podesimo osjetljivost (slika 2). U glavnom meniju Recognita nas obavestava iako će u kakvom će formatu biti zapisan prepoznati tekst (opcija «Format» - biramo među formatima različitih tekst editora) koju će masu slova (azbuku) Recognita identifikovati (opcija «Enabled» - ne raspolaganju ima-

mo tako reći, sve evropske azbuke) i koji standard kodova upotrebljava. I pri zapisu («Set») - ovdje možemo da podesimo YUASCII Recognita nam takođe pokazuje koje će stablo upotrebljavati kod identifikacije teksta (opcija «Tree») da li svuje još ranije određeno ili neko koje je kreirala za vreme učenja.

Uz glavni meni imamo na raspolaganju još tri «Windows» - «Options» i «Statistics». Prozorima određujemo one delove strana koje program treba da uzme u obzir kod identifikacije, odnosno za koje ne treba da maruje tako eliminišemo delove (slika 3) Recognita će doduše sama razlikovati slike od teksta, ali to spor proces, uvek nije ni 100% pouzdan pa je zato bolje programu malo pomoći. U meniju opcija (slika 4) biramo azbuku teksta format zapisa i kodni standard a osim toga još stablo identifikacije. Možemo takođe da odredimo znak za znakove koji nedostaju («Missing Symbol») i znak za neidentifikovane znakove («Rejection symbol»).

Opcijom «Read» u glavnom meniju pokrećemo identifikaciju. Pokazace nam se ekran bez teksta samo sa slikom bicikliste koji neumorno okreće pedale. Već posle nekoliko sekundi pokazuje se prvi red prepoznatog teksta i u minutu još svi drugi (računar 286/16). Neidentifikovana slova program označuje crnim kvadratićem. Posle identifikacije Recognita nas pita u koju datoteku želimo da zapišemo prepoznati tekst izborom «Statistics» a glavnom meniju saznajemo koliko je znakova, odnosno reči program prepoznao, koliko ih nije prepoznao i kakva je bila pouzdanost prepoznavanja. Ta statistika je na žalost lažna jer program tu ne može da broji znakove koje ■ prepoznao pogrešno.

Ako identifikovanim tekstom nismo zadovoljni možemo da se odlučimo među više mogućnosti. Prva

OCR editor teksta rečnik i jezički analizator američka firma OmniPage

Upotrebljivost programa za OCR

Program je upotrebljiv za OCR ako je prepoznavanje teksta u njemu brzo i pouzdanije od prethodno. U računar teksta datoteka grafičkom koda bitim 480 odnarača u minutu do je pet minuta za učitavanje kasnije stranicu. Zbog se brzo pouzdanost od 99.9% odsto dve greške na hiljadu znakova odnosno jedna na hiljadu stranu. Vrhunski programi Recognita i OmniPage postaju u istovremeno uložena i štampanje knjiga, prepoznavanje 49 znakova ali u stvarnosti tako uspešno testiranje po sloje. Uba programi su i zavodljivo brža. Kućanu stranu (2000 znakova) na računaru prepoznaju procesorom 286/33 za otprilike pola minuta. To vreme procesor 286

najmanje udvostručava. Treba imati u vidu i vreme pripremanje teksta i proveravanje prepoznatog teksta, tako da prepoznavanje tekstova ima da potraje i treba imati dosta lake razloge za nabavku kvalitetnog optičkog čitača i programa (uključno 5000 do 6000 DEM). Ne programi za OCR koji se mogu dotiči stručnim čitačima po cenji 300 do 500 DEM slobodno zaboravite, jer za nešto više od poligrafičari uopšte ne dolaze u obzir (preciznost im je 30 do 90 odsto, brzina 2 do 3 minute). U ovoj razmazu vreme, kada računari postaju sve snažniji, teško je prihvatiti da 30 sekundi za prepoznavanje jedne stranice teksta predstavlja nekakvo izuzetno dostignuće, jer niko i ne pomišlja na složenost postupaka koji to omogućavaju, zaboravljajući pri tome da je još do pre nekoliko godina prepoznavanje teksta bilo moguće samo specijalno podešenim računarima (sa dodatnom memorijom i procesorima). O tehnici OCR sve se više govori i ima sve više javnih programa za OCR, što mo-

že prosečnog korisnika računara da navede na pomisao da je to jednostavan, već potpuno rešen problem. Po mom mišljenju, programi za OCR postaju opetie upotrebljivi tek kad program prepoznavanja padne pod režimo: bat sekundi (puna pod 500 DEM) i kad preciznost prepoznavanja u te žim uslovima dostigne najmanje 99,95 odsto (jedna greška po kucanoj strani). Računar sa tako brzim procesorom danas još ne postoji, bar u kancelarijskoj sredini.

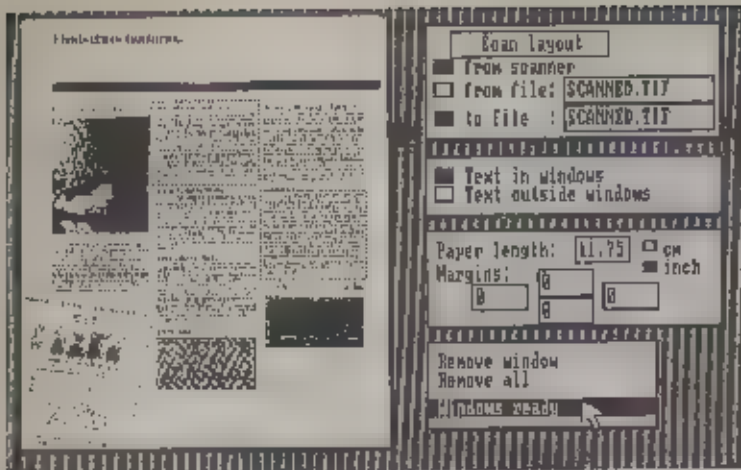
Kakav program kupiti?

Ako se odlučujete za sistem za optičko prepoznavanje znakova, obratite pažnju na sledeće osobline programa:

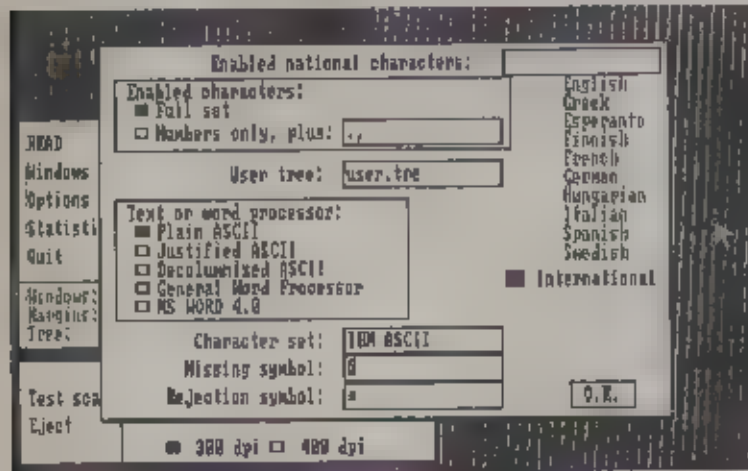
1. Brzina prepoznavanja mora da iznosi najmanje 30 znakova u sekundu (to sigurno neće biti slučaj računarom XT).

2. Preciznost prepoznavanja mora da bude najmanje 99 odsto, u protivnom se tekst bitni neupotrebljiv.
3. Da li je sistem sposoban za učenje novih znakova?
4. Da li prepoznaje i grafičke datoteke?
5. Da li prepoznaje razne dimenzije i tipove slova?
6. Koje optičke čitače podržava?
7. U koje formate može prepoznati tekst da zapiše?
8. Koja proširenja memorije podržava?
9. Da li automatski prepoznaje i prekače grafičku u tekst?

Dok se danas na tržištu već mogu naći mnogi javni i javni programi za OCR (300 do 500 DEM), koji se najčešće dobijaju u paketu sa ručnim optičkim čitačem, izbor profesionalnih programa veoma je mali. Pored već pomenutih OmniPage i Recogniti on, prema trenutno dostupnim informacijama u obzir dolazi još samo američki ReadRight.



Slika 3. Određivanje prostora



Slika 4. Opcije

je da ručno popravimo greška nekim tekst editorom; to je sporo i ne-pouzdatno obavljanje posla. Druga mogućnost je da stavimo tekst na proveravanje programu s rečnikom (spellchecker) koji će automatski potražiti sve sumnjive reči. Treća mogućnost je da pokušamo da promenimo osetljivost optičkog čitača, i, konačno, od programa možamo da zahtevamo da za vreme prepoznavanja uči. To se odvija tako da nas program pita za svako slovo

koje nije ubeđen da ga je pravilno prepoznao i svako koje uopšte ne može da prepozna. Na slici 5, gde je slovo «m» odštampano debelo, program je slovo doduše identifikovao, ali nije bio ubeđen da je identifikacija pravilna, pa nas je zato pitao za potvrdu da li to slovo da nauči. Od sada će takav, nešto masnije odštampani «m» prepoznavati bez problema. Ako neko slovo ne može da prepozna, onda mu to slovo treba reći. Tako možemo program da

Na sajmu System u Minhenu došli smo i nekoliko informacija o tržištu uređaja za optičko prepoznavanje tekstova sledećih nekoliko godina (analizu tržišta obavio je Institut za marketing Frost & Sullivan iz Frankfurta). Po mišljenju Instituta, prodaja optičkih čitača treba tokom naredne tri godine da raste godišnjom stopom od 60 do 70 odsto, jer je takav trend započeo još 1988. godine, dok će naglim porastom kvaliteta proizvoda - optičkih čitača i programa za prepoznavanje - još da se nastavi. Ako su 1988. godine prodati optički čitači i ali u desetine hiljada, posle 1992. godine treba da idu u stotine hiljada. Međutim, današnji nivo kvaliteta još ne zadovoljava, tako da će za ostvarenje ovih predviđanja morati da se unapredi. Praktična preciznost prepoznavanja tekstova boljih programa danas iznosi između 97 i 99 odsto, što je još daleko od praktične upotrebljivosti, jer u prepoznavanju kućane stranice teksta znači i do 100 grešaka. Najviše navodno očekavaju

računarske arhitekture na osnovi neuronskih mreža i paralelnog procesiranja, jer će jedino u njima moći da se izvode programi za OCR nove generacije (oni treba da prepoznaju i rukopis). Oprema (čitači) zapravo već danas odgovara kriterijumima kvaliteta, dok u programima mogu da se očekuju velike kvalitetne izmene do konačne i željene stoprocentne preciznosti prepoznavanja i odgovarajuće brzine za praktičnu primenu. Iscrpnu studiju Scanning and Optical Recognition Equipment Market in Europa zainteresirani mogu da naruče od Frost&Sullivan Instituta, po ceni 3300 dolara.

DHL »express«

Čitaocima dugujemo izvinjenje, jer smo ovaj test najavili još za prošlogodišnji letnji broj Moj mikro, s obzirom na to da nam je proizvođač SZKI iz Budimpešte obećao novu verziju 1.1 za poč-

naučno i na neka povezana slova kao što je, recimo, par «fi» u nekim grafičkim slogovima.

Kad bismo morali da ocenjujemo svojstva Recognite, ocenama od 1 do 5, evo kako bi prošla:

- priručnik	5
- instalacija	5
- ljubaznost okoline Windows	5
- ljubaznost okoline DOS	3
- podrška optičkih čitača	-5 (na podržava ručne)
- podrška tekst editora	+5 (skoro suviše egzotika)
- različita azbuke (aboceda)	-5 (evropska osim ćirilica)
- različiti kodni standardi	5
- brzina prepoznavanja	5 (danas brže nije moguće)
- lačnost prepoznavanja	4
- cena	2

Može se reći da je Recognita izvanredno dobar program za današnji stepen razvoja tehnika OCR, mada to još nije ono što bismo želeli, naročito u pogledu brzine. Test tekstove, priložene priručniku, program je naravno, prepoznao 100%. Isto tako nije bilo problema sa izvanredno lepom štampom luksuznih knji-

mo preko 95% (u prvom redu zbog loše štampanih slova). Verovatno bi sa profesionalnim optičkim čitačima postigli još bolji rezultati, ali o tome drugi put.

Slika 5. Učenje



tak jula. Pošiljku je zaista poslao 4. jula i da bi što brže stigla, upotrebio je ekspres dostavu DHL, koja u pravilu stiže za dva dana u bilo koji deo Evrope. Međutim, mi smo tek 15. avgusta iz ljubljanskog sedišta DHL u Javnim skladištima dobili obavještenje da se »navodno negde u Zagrebu nalazi neka pošiljka za nas« (kasnije smo proverili i utvrdili da je adresa bila apsolutno tačna) i da treba sami da se raspitamo u zagrebačkom sedištu DHL. Posle nekoliko razgovora sa Zagrebom (nisu znali šta da rade s pošiljkom) i ljubljanskim centralom, poslao nam je za rukom da se paket 24. avgusta uputi u Ljubljanu, odnosno na Brnik, gde ga je preuzela špedicija ŽG Feraped. Tu su nastale nove komplikacije: prvo sa računom, na kojem nije pisalo da je to primerak, namenjen testiranju, već je stajala samo vrednost od 20 USD, ali kada je iz Budimpešte (25. septembra) stigao telefaks s izjavom da je to zalista test primerak vrednosti 20 USD, izgubljen je tovarni

list (nije nam jasno da li je za to bio kriv DHL ili špediter). Paket je konačno ocarinjani 10. oktobra, a mi smo ga preuzeli tek početkom novembra na Brniku, iako bi morao da bude dostavljen u DHL, jer mi nismo svakih pet minuta zvali aerodrom, a špediter, odnosno kurirska služba nisu smatrali da bi ipak trebali da nas obaveste da je paket ocarinjani. Isporuke ove ekspresne poslata pošiljka potrajala je puna četiri meseca.

Ukupni troškovi za paket sa deklarisanom vrednošću 20 USD iznosili su 1591,70 dinara. Od toga je carina bila 35 dinara (naplaćeno nam je previše, jer je trebalo cariniti samo vrednost dve diskete, a to je najviše 2 USD, što bi bilo 3,5 dinara), dok su drugo bilo: takse marke, telefon, špeditersko posredovanje, carinsko posredovanje, taksa, provizija, skladište, porez za JNA. Ali, to je već druga priča (vodl članak Uvoz programe poštom). Naravno, kurirska služba DHL nije bilo ni na kraj pameti da nam uputi bilo kakvo izvinjenje.

AVTOTECHNA

Produktions- und Warenhandelsges. m.b.H.

St. Veiterstr. 41, Celovec (Klagenfurt), Austrija
 Telefon: 9943 463 50578
 Telefax: 9943 463 50522
 Informacije u Ljubljani:
 (061) 323 755 i (061) 329 067

Računarske komponente ■ konfiguracije:

Kućiste AUVA baby AT/200 W, CPU AUVA 286-12 MHz, SLINTAC LIM 4.0, 1 Mb RAM Intel, hercules/štampač kartica AUVA, kontroler AUVA AT-bus, floppy 1,2 Mb TEAC, tastatura AUVA.US 102 kljk ■ kontaktima cherry, 14" monitor AUVA, čvrsti disk seagate 45 Mb/28 ms

DEM 1.356.- netto, bez MWST

Računare prodajemo po komponentama:

Kućiste AUVA baby sa 200 W uredajem za napajanje	DEM 165.-
Kućiste AUVA, mini stub sa 200 W uredajem za napajanjem	220.-
Kućiste AUVA stub ■ 220 W uredajem za napajanjem	299.-
CPU ploča AUVA AT 286/12 MHz, one chip	190.-
CPU ploča AUVA AT 286/16 MHz, NEAT	315.-
CPU ploča AUVA 386SX/16 MHz	699.-
CPU ploča AUVA 386DX/25 MHz/32Kb cache	1.320.-
RAM 1 Mb - 80 ns (18 × 41256/80)	108.-
RAM 2 Mb - 70 ns (18 × 411000/70)	252.-
RAM SIMM 256 K/80 ns	37.-
RAM SIMM 1 Mb/70 ns	120.-
Herkules/printer kartica AUVA	28.-
VGA color kartica AUVA, 16-b/512 K/1024 × 768	199.-
2× serijski interfejs AUVA	25.-
2× ser./1× paral. interfejs AUVA	34.-
2× ser./1× paral./game interfejs	36.-
FDD/HDD kontroler AUVA AT-bus	38.-
FDD/HDD kontroler AUVA MFM, 1:1	106.-
FDD/HDD kontroler seagate SCSI	99.-
FDD/HDD kontroler NCL ESDI	399.-
Tastatura AUVA US 102, kljk, cherry	69.-
14" monitor AUVA, c/b ili ambera	174.-
14" monitor VGA AUVA, kolor, 1024×768	675.-
Monitor NEC 3D	1.490.-
Čvrsti disk Seagate ST 157A, 45 Mb/28 ms	445.-
Čvrsti disk Seagate ST 1096N, 83 Mb/25 ms	685.-
Čvrsti disk Seagate ST 1201A, 177 Mb/15 ms	1.480.-
Čvrsti disk Seagate ST 1239A, 211 Mb/15 ms	1.480.-
Čvrsti disk NEC D3142, 42 Mb/25 ms	499.-
Čvrsti disk NEC D3741, 44 Mb/25 ms, AT-bus	532.-
Mis Genius GM+	67.-
Streamer 60 Mb colorado	778.-
Diskete 1,2 Mb, 5,25" BASF extra	2,40
Štampač Epson LX-400, 9 igl., A4	399.-
Štampač Epson LQ-400, 24 igl., A4	690.-
Štampač Epson LQ-1010, 24 igl., A3	1.180.-
Laserski štampač OLUME crystal print	3.100.-
Laserski štampač hewlett-packard HP-III	3.600.-

Garancija: 1 godina, ■ Ljubljani.

PRODAJA ZA DINARE:

**GROSISTIČKA PRODAJA
 – POSEBNI ARANZMAJI!**

TECHNOS d.o.o.

Računarska oprema – servis

Titova 25c

61000 Ljubljana

tel.: (061) 323-755, 329-067

fax.: (061) 329-067

RAČUNARI AUVA

AUVA 230/12 din 22.900,-

Konfiguracija:

kućiste slim, 200 W, CPU 286-12 MHz Sultac LIM 4.0, 1 Mb RAM, monohrom, grafička kartica/paralelni interfejs, ser./par. Interfejs, kontroler 1:1, gipki disk TEAC/Mitsubishi 1.2 Mb, čvrsti disk 42 Mb Seagate ST 251-1, tastatura US 101 kljk, monitor 14" c/b, originalni Microsoft DOS 3.3 sa knjigom, priručnik za rad na računaru. Svi računari testirani ■ posebnim postupkom od 48 časova.

AUVA 250/16 din 24.290,-

Konfiguracija:

kućiste baby, CPU 286-16 MHz NEAT, ostalo isto kao gore.

AUVA 900/16 din 26.526,-

Konfiguracija:

CPU 386X-16 MHz, ostalo isto kao gore.

Računar 286-12 po komponentama: din 17.750,-

Konfiguracija:

kućiste baby/200 W, CPU 286-12 sultac, RAM 1 Mb, hercules/port za štampač, kontroler AT-bus, floppy 1,2 Mb TEAC, čvrsti disk seagate 45 Mb/28 ms, tastatura US 101, 14" crno-beli monitor.

Na raspolaganju su i:

Štampači epson, crtači roland, monitor EIZO, čvrsti diskovi NEC, seagate, micropalis, gipki diskovi TEAC, integrisane kula inte, računarske komponente AUVA.

Navedene cene sa bez poreza na promet.

Molimo, pozovite nas, cene se menjaju!

Objekti u računarskoj grafici (TP 5.5)

DUŠKO SAVIĆ

Objektno programiranje na velika vrata ulazi u programske jezike i komercijalne PC pakete. To je najbolji način programiranja za koji danas znamo. Objektno programiranje obuhvata tri grupe jezika. U prvoj su čisti objektni jezici, npr. Smalltalk/V i Actor. U njima je bukvalno sve podvedeno pod pojam objekta: i prozori su objekti, i editori, i miš. U drugoj grupi su «skoro» objektni jezici (C++), tj. jezici u kojima je bolje i jednostavnije upotrebljavati objekte nego koristiti klasične mogućnosti jezika. U trećoj grupi su hibridni jezici, npr. Turbo Pascal 5.5 i QuickPascal, u kojima možemo sve raditi i bez objekata.

Turbo Pascal je odavno, od verzije 3.0, najpopularniji pojedinačni programski jezik na PC računarima. O tome svedoči i preko 30 knjiga, ne računajući hiljade programa koje su napisane tim dijalektom. Verzija 4.0 Turbo Pascala dodala je i izv. jedinice, koje su u ostalim programskim jezicima poznate kao moduli. U toj verziji Turbo Pascal je skoro dostigao snagu Module-2. Sledeća verzija, 5.0, dodala je ugrađeni debager (debugger), ali nekih posebnih dodataka u jeziku nije bilo. Od verzije 5.5 Turbo Pascal podržava i objekte. Koristi je dvostruko: Turbo Pascal hvata korak sa vremenom, i istovremeno omogućava dvomilionskoj grupi svojih poklonika da se oprobaju u ovoj novoj vrsti programiranja.

Verovatno najbolja osobina ove verzije je neagresivnost. Ko ne želi da učini nove stvari, može mirne duše ignorisati nova četiri službene reči kroz koje se realizuje objektno programiranje na Turbo Pascalu 5.5. Svi prethodno napisani programi rade kao i pre, nema potreba pri prelasku na novu verziju. Ipak, objektno programiranje je isuviše zgodno i moćno da bi se na duži rok tek tako moglo ignorisati. Rešava neke probleme koji su toliko ukorenjeni u tradicionalni način programiranja da većina programera misli da je programiranje jednostavno teško samo po sebi i da će zauvek ostati takvo. Tipičan programer rezonuje i ovako: «Opet buka. Sačekajmo da i ovaj talas prođe... moj program je već dugačak 5000 linija i ko će s tim da se bori. Nemam ja vremena da učim iznova samo da bih se ponovo razočarao kao onomad sa strukturiranim programiranjem. Poverovaću da je objektno programiranje dobro tek kad njegove zagovornike vidim kako voze maserati!»

Ironija je u tome što se ljudi plaše promena i unapred odbacuju upravo onu vrstu programiranja koja im najviše može pomoći u borbi protiv promena – objektno programiranje. Bilo bi naivno očekivati da će programeri preći na objektno programiranje sami od sebe. Cilj ovog teksta je da pokaže kako se objekti mogu koristiti u jednoj oblasti koja je svima dobro poznata – u računarskoj grafici. Nećemo se baviti ni teorijom ni sintaksom, pretpostavićemo da imate originalna priručnike ili neku knjigu detaljno opisuje opisom Turbo Pascal 5.5. Takođe pretpostavljamo da znate definiciju matrica i osnovne operacije sa njima. Krajnji cilj je da napišemo klasu koje će crtati grafičke objekte u dvo- i trodimenzionom prostoru. Prvo ćemo matematičke matrice predstaviti kao objekte, a nastaviti ćemo sa raznim funkcijama, linijama, poligonima, površima i ostalim očiglednim objektima. Time istovremeno odgovaramo na često pitanje početnika – kako prepoznati objekat? Odgovor je vrlo prost: sve može biti objekat, samo je pitanje da li to vodi postavljenom cilju.

Klasa sadrži opšte osobine objekta a objekat je konkretizacija klase.

Matricne operacije u Turbo Pascalu

Turbo Pascal nema matrice kao osnovni tip podataka, što znači i da nema odgovarajuće operacije nad njima. Klasu matrica moramo sami napisati. To se može izvesti na nekoliko načina. Iako je to prva klasa u ovom tekstu, postavljamo pitanje kojim počinje pisanje svakog objektnog programa: «Gde umetnuti novu klasu u hijerarhiju već postojećih klasa?» Isto to, samo malo drukčije: «Od kojih klasa će nova klasa, Matrix, naslediti najviše korisnih osobina?» Ta pitanja su retorička jer Turbo Pascal nema neku klasu iz koje bi izvirale sve ostale klase. (Poređenja radi, u jeziku Smalltalk/V svi objekti nasleduju objekat koji se zove Object.) Klasa Matrix će zato biti potpuno nova, a kao što ćemo videti, nju će nasledivati mnoge druge klase.

Matrica je uređena m-torka uređenih n-torki, tj. niz nizova. U pascalu se to na klasičan način izražava ovako:

```
TYPE
  rows = ARRAY [1..10] OF REAL;
  matrix = ARRAY [2..34] OF rows;
```

Tip rows je niz od deset brojeva tipa real, a matrix je niz promenljivih tipa rows. Problem sa ovom reprezentacijom matrica je da su granice fiksirane već tokom prevodenja. Ovak pristup definisanju matrica je zato neupotrebljiv za definisanje matrica kao opšte klase. Bolje je stvoriti prosti niz i izračunavati adrese matricnih elemenata u njemu. Takav niz stvaramo dinamički, tj. tokom izvršavanja programa, pozivom standardne naredbe Turbo Pascala, GetMem. Tu pretpostavljamo da ukupna dužina matrice ne sme biti

veća od dužine segmenta, tj. mora biti manja od 64 K. To je prihvatljivo ako se ima u vidu cilj ovog teksta – crtanje grafičkih objekata.

Mogućnosti su i drugi pristupi internom upravljanju memorijom za klasu Matrix. Postoje tzv. virtualne matrice, kod kojih je najveći broj elemenata u datoteci na disku, a memorija služi kao «prozor» kroz koji se gleda na celu matricu. Matrice koje se uglavnom sastoje od nula (takve matrice po pravilu dobijaju se rešavanjem parcijalnih diferencijalnih jednačina) zahtevale bi neki treći način upravljanja memorijom. Lepota objektnog programiranja je u tome što različite vrste matrica možemo napisati kasnije, a da algoritmi koji koriste matrice uopšte ne promene!

Klasa Matrix nije savršena, ali je sasvim dovoljna i za potrebe računarske grafike i kao začetnik nekih novih matricnih klasa.

Klasa Matrix

Evo kako izgledaju naredbe za klasu Matrix, iz datoteke MAT.PAS:

```
{c++}
unit Mat;
interface
const stringLength = 255;
type
  dataType = real;
  typeArray = array [1..1] of dataType;
  mPtr = ^typeArray;
  matRecPtr = ^matRec;
  matRec = record
    m : mPtr;
    size : word;
    rows,
    columns : Integer;
    name : string[stringLength];
  end;
  Matrix = object
    mat : matRecPtr;
    procedure Init(i, j : Integer; m : string);
    procedure done;
    function name : string;
    function loc(i, j : Integer) : longInt;
    procedure wr;
    procedure zero;
    procedure identify;
    procedure changeName(n : string);
    procedure identify;
    procedure plus(mat1 : Matrix);
    procedure mult(m : Matrix; n : Matrix);
    function get : matRecPtr;
    procedure put(value : dataType; i, j : Integer);
    function getColumns : Integer;
    function getRows : Integer;
    function at(i, j : Integer) : dataType;
  end; { Matrix }
```

Implementation

```
procedure Matrix.changeName;
begin
  mat.name := n;
end;
procedure Matrix.identify;
begin
  write(mat.name);
end;
function Matrix.getColumns;
begin
  getColumns := mat.columns;
end;
function Matrix.getRows;
begin
  getRows := mat.rows;
end;
procedure Matrix.Init;
begin
  new(mat);
  mat.rows := i;
  mat.columns := j;
  mat.size := (i * j) * SizeOf(dataType) + SizeOf(MatRec);
  mat.name := n;
  GetMem(mat.m, mat.size);
```

```

end;
procedure Matrix.done;
begin
  dispose(mat);
  FreeMem(mat^m, mat^size);
end;
function Matrix.name;
begin
  name := mat^name;
end; { name }
function Matrix.loc;
begin
  loc := SizeOf(dataType) * (( - 1) * mat^rows + 1) + SizeOf(MatRec);
end; { loc }
procedure Matrix.wr;
var l, j : integer;
begin
  writeln(mat^name);
  for l := 1 to mat^rows do begin
    for j := 1 to mat^columns do
      write(' ', at(i,j):13:4);
      writeln;
    end;
  end;
end;
procedure Matrix.zero;
var l, j : integer;
begin
  for l := 1 to mat^rows do
    for j := 1 to mat^columns do
      mat^m[loc(l,j)] := 0;
    end;
  end;
end;
procedure Matrix.identify;
var l : integer;
begin
  zero;
  for l := 1 to mat^rows do
    mat^m[loc(l,l)] := 1;
  end;
end;
procedure Matrix.plus;
var l, j : integer;
a, b : dataType;
begin
  for l := 1 to mat^rows do
    for j := 1 to mat^columns do
      put(at(l,j) + mat1.at(l,j), l,j);
    end;
  end;
end;
function Matrix.get;
begin
  get := mat;
end;
procedure Matrix.put;
begin
  mat^m[loc(l,j)] := value;
end;
function Matrix.at;
begin
  at := mat^m[loc(l,j)];
end;

procedure Matrix.mult;
var l, j, k : integer;
a, b, c : datatype;
begin
  for l := 1 to m.getRows do
    for j := 1 to m.getColumns do begin
      put(0,l,j);
      for k := 1 to n.getColumns do begin
        a := m.at(i,k);
        b := n.at(k,j);
        c := at(i,j) + a * b;
      }write('a ', a:13:4, ' b ', b:13:4, ' a*b ');
      writeln(a*b : 13:4, ' ', i:2, ' ', j:2, ' k ', k:2);}
      put(c, l, j);
      end; { k }
    }writeln('result ', at(i,j):13:4);}
    end; { j }
  end; { mult }
end. { Mat }

```

Matricni elementi su tipa `dataType`, koji ovde nije ništa drugo do tip `real`. Slog `matRec` sadrži sledeća polja:

- , pokazivač na niz u kojem je matrica,
- ize, broj bajtova koji zauzimaju elementi matrice,
- rows, broj redova u matrici,
- columns, broj stubaca u matrici,

`name`, ime matrice; korisno kod testiranja, u sprezi sa metodom `wr`. Promenljiva `rows` i `columns` su matricne dimenzije. Procedura `init` (koja nije konstruktor, jer je ovo začetnik buduće hijerarhije matricnih klasa) postavlja ove vrednosti dinamički. Nuzprodukt procedure `init` je vrednost u `size`, što je i broj bajtova koje u memoriji rezervišemo za matricu. Ako je `size > 65521`, efekti definisanja matrice u programu nisu poznati. Poruka poput `init(333,666)`, kojom se zauzima $333 \times 666 \times 4 = 887112$ bajtova, ■ stvari radi, ali nije jasno šta se dešava kada program počne da pristupa nepostojećim delovima memorije. Parametre poruke `init` treba kontrolisati, bilo kod zadavanja ili ugrađivanjem zaštite u metod `Matrix.init`.

«Magična konstanta» ■ ovom slučaju iznosi 65521 a ne svih 65535 kao što bi se moglo očekivati. Sniženje granice potiče od samog Turbo Pascala, odnosno, od načina na koji on zauzima dinamičku memoriju (heap memory).

Bitno ■ rezervisati prostor za ceo matricni slog `matRec` a ne samo ■ niz sa elementima matrice. Da nismo dodali `SizeOf(matRec)` u metod `Matrix.loc` svi programi koji koriste klasu `Matrix` ponašali ■ se čudno, zbog prepiljanja podataka po istim memorijskim adresama. Pri tome do grešaka može doći u najneočekivanim trenucima, npr. posle naredbe `writeln` ili posle neke druge standardne naredbe Turbo Pascala.

Procedura `init` zauzima prostor pomoću `GetMem`; rezervisani prostor kasnije oslobađamo ■ proceduri `done` pozivom procedure `FreeMem`. (Primitimo da ovde `done` nije destruktor, kao što ni `init` nije konstruktor – `Matrix` nije virtualna klasa.) Prvi parametar `l` za `GetMem` i za `FreeMem` je pokazivač na niz a drugi parametar ■ dužina niza. Radi se ■ triku: prevodilac zauzima prostor kao da se drugi i prvi argumenti slažu.

Naredbe `GetMem` i `FreeMem` treba zvati ovako:

```

GetMem(typeArray, SizeOf(typeArray));
FreeMem(typeArray, SizeOf(typeArray));

```

ali to ne moramo slepo poštovati. Znaćemo ove potprograme sa lažnim pokazivačima, ali sa ispravnom dužinom; to radi, pod uslovom da memoriji pristupamo u skladu sa ispravnim tipovima podataka. Zato uvodimo metod po imenu `loc(l,j)`. On čini javno ono što prevodilac svu vreme radi tajno – izračunava tačne adrese za tip podataka koji je obeležen sa `dataType`.

Time gubimo na još jednoj sintaksnoj pogodnosti: umesto da elementu matrice pristupamo pišući `m[i,j]`, sada moramo pisati `m^loc(l,j)` jer je `m` linearan niz u okviru kojeg `loc` vraća adrese. Preciznije, moramo pisati `m^loc(i,j)` jer je `m` ■ suštini pokazivač na `typeArray`. Taj je niz takođe deo sloga `matRec`. Promenljiva `mat` je jedina promenljiva u okviru klase `Matrix` i ima tip `matRec`. Zato nije dovoljno pisati `m^loc(i,j)` da bi se došlo do elementa matrice. Potpuni izraz glasi `mat^m^loc(l,j)`, čime označavamo element objekta koji prima poruku. Ovo izgleda komplikovano, ali do takvih izraza se po pravilu dolazi korak po korak, a kada ih jednom napíšemo – možemo zaboraviti na njih.

Ako u nekom kasnijem trenutku maksimalna dužina pojedinačne matrice mora biti duža od 84 ■ treba samo izmeniti metod `loc`. Van klase `Matrix` ne treba upotrebljavati `loc`, a i svi metodi ■ toj klasi do matricnih elemenata dolaze samo preko `loc`. Primitimo da je u Turbo Pascalu potpuna lokalnost metoda kao što je `loc` stvar programerove samodisciplinacije – u drugim objektnim jezicima do takvih metoda se uopšte ne može doći osim ako programer to izričito ne dozvoli.

Metodi klase matrix

Metod `init` prvo formira novi pokazivač na `mat`, postavlja vrednosti promenljivih, zatim poziva `GetMem` da bi zauzeo prostor za `mat`. Metod `done` je suprotan: sa `dispose` oslobađa prostor zauzet za `mat`, a sa `FreeMem` oslobađa memoriju rezervisanu za matricne elemente. Metod `loc` je standardan način da se dođe do pojedinačnog elementa matrice u linearnom nizu. Metod `wr` prvo piše ime matrice a zatim i samu matricu. Time se spasavamo velikog broja `writeln` naredbi u glavnom programu. Metodi `zero` i `identify` stvaraju neutralne matrice za sabiranje i množenje. Metodom `get` dolazimo do broja u matrici. Sa `put` upisujemo broj `value` na mesto `i, j`. Metod `at` vraća matricni element na poziciji `(i,j)`. Dobro ■ bilo da svi drugi metodi klase `Matrix` do matricnih elemenata dolaze isključivo preko `put` i `at`. U objektnom programiranju na Turbo Pascalu znak `:=` za operaciju pridodeivanja ne postoji.

Metodi `changeName` i `identify` postavljaju i ispisuju ime matrice ■ promenljivoj `name`, ali bez samih matricnih elemenata. Konačno, metodi `plus` i `mult` su binarne aritmetičke operacije nad matricama: `plus` sabira dve matrice, a `mult` ■ množi, prethodno proverivši da li je matricno množenje izvodljivo za zadate argumente. Ako nije, šalje poruku na ekran.

Često se postavlja pitanje koji sve metodi treba da budu u klasi. Najbolje je da programer sam počne da upotrebljava svoje klase. Na početku je metod `get` izgledao kao sasvim upotrebljiv način da se dođe do podataka u matrici. Već posle prvog test programa za ovu klasu osetila se potreba za metodom `at`, koji radi suprotno od metoda `put` – vraća podatak ■ matrica.

Naravno, možemo dodati veliki broj drugih matricnih operacija, ali pogledajmo prvo kako ove rade. Program `mmMat` glasi:

```

{mm+}
{mm+}
program mmMat;
uses CrI, Mat, TCUtil;

```

```

var
  g : word;
  k, m, n : Matrix;
begin
  ClrScr;
  writeln('mmMat prvi ekran - Demo za Matrix'); writeln;
  n.Init(4,4, 'n'); n.Identity;
  with m do begin
    WriteLn('Prvi nultu matricu od 6 redova i 3 stupca');
    WriteLn; Init(6,3, 'm'); zero; wr; done;
    WriteLn('Menja je m matricu 4 puta 4, ispunjenu brojevima');
    m.Init(4,4, 'm');
    put(1,1,1); put(2,1,2); put(3,1,3); put(4,1,4);
    put(4,2,1); put(4,2,2); put(0,2,3); put(4,2,4);

    put(7,3,1); put(5,3,2); put(7,3,3); put(2,3,4);
    put(8,4,1); put(9,4,2); put(1,4,3); put(0,4,4);
    wr;
    WriteLn('Dodajemo jedinicnu matricu');
    plus(n);
    changeName('m + n'); wr;
    g := GetKey;
    ClrScr;
    WriteLn('mmMat drugi ekran - matricno mnozenje');
    WriteLn; WriteLn('Matrix m'); WriteLn;
    wr;
    n.changeName('n m jedinicna matrica'); n.wr;
    k.Init(m.getRows, n.getColumns, 'k = m * n');
    k.mult(m, n);
    WriteLn;
    WriteLn('Rezultat mora biti (numericki) isti kao m');
    WriteLn;
    k.wr; m.done; n.done; k.done;
  end; { with m }
end. { mmMat }

```

Razultati programa mmMat su na slikama (1) i (2). Promenljive k, m, n su objekti klase Matrix. Matricu m prvo deklariramo i izdajemo na ekran kao matricu sa šest redova i tri stupca. Zatim sa m.done brišemo m kao matricu dimenzija 6 x 3 i redefinišemo je kao matricu 4 x 4. Njene elemente poslavljamo nizom put metoda. Matrica n je jedinična i dodajemo je matrici m. Pošto važi naredba with m do, metod plus(n) znači isto što i m.plus(n). Rezultat ispisujemo metodom wr, koji je - istih razloga - identičan sa m.wr.

Funkcija GetKey iz jedinice TCUnit zauzima izvršavanje programa dok se očita novi pritisak na tastaturu.

mmMat prvi ekran - Demo za Matrix

Prvi nultu matricu od 6 redova i 3 stupca

m	0.0000	0.0000	0.0000
	0.0000	0.0000	0.0000
	0.0000	0.0000	0.0000
	0.0000	0.0000	0.0000
	0.0000	0.0000	0.0000
	0.0000	0.0000	0.0000
	0.0000	0.0000	0.0000

Menja je u matricu 4 puta 4, ispunjenu brojevima

m	1.0000	2.0000	3.0000	4.0000
	4.0000	4.0000	0.0000	4.0000
	7.0000	5.0000	7.0000	2.0000
	8.0000	9.0000	1.0000	0.0000

Dodajemo jedinicnu matricu

m + n	2.0000	2.0000	3.0000	4.0000
	4.0000	5.0000	0.0000	4.0000
	7.0000	5.0000	8.0000	2.0000
	8.0000	9.0000	1.0000	1.0000

Slika 1. Rezultati prvog dela programa mmMat.

Drugi ekran programa mmMat demonstrira matricno mnozenje.

mmMat drugi ekran - matricno mnozenje

Matrix m				
m + n	2.0000	2.0000	3.0000	4.0000
	4.0000	5.0000	0.0000	4.0000

	7.0000	5.0000	8.0000	2.0000
	8.0000	9.0000	1.0000	1.0000
n je jedinicna matrica	1.0000	0.0000	0.0000	0.0000
	0.0000	1.0000	0.0000	0.0000
	0.0000	0.0000	1.0000	0.0000
	0.0000	0.0000	0.0000	1.0000

Rezultat mora biti (numericki) isti kao m

k = m * n	2.0000	2.0000	3.0000	4.0000
	4.0000	5.0000	0.0000	4.0000
	7.0000	5.0000	8.0000	2.0000
	8.0000	9.0000	1.0000	1.0000

Slika 2. Rezultati drugog dela programa mmMat.

Kvadratne matrice - klasa SquareMatrix

Sada je sasvim lako dodati matricno oduzimanje, mnozenje matrice skalarom, kopiranje stubaca i redova, brisanje itd. Medutim, analogija prestaje da važi kod matricnog deljenja. Matricno mnozenje nije komutativno što znači da mnozenje matrica zavisi od rasporeda argumenata. Ne postoji jednoznačno matricno deljenje u opštem slučaju. Zato moramo dodati novi uslov, tj. naći podskup opštih matrica za koje postoji jednoznačno deljenje ili neka slična operacija. Dodajemo uslov da dimenzije matrica moraju biti međusobno jednake, i to su tzv. kvadratne matrice. Za takve matrice uvodimo pojam inverzne matrice. Matrica pomnožena svojom inverznom matricom (ako ova postoji) daje jediničnu matricu.

Kvadratna matrica je isto što i obična matrica, ali tako da je m=n. Klasa kvadratnih matrica će zato biti potklasa klase Matrix, nazovimo je SquareMatrix. Pretpostavimo da je u datoteci MMSQMAT.PAS:

```

unit SqMat;
{ Kvadratne matrice }
interface
uses Mat;
type
  SquareMatrix = object(Matrix)
  constructor Init(l : Integer; n : string);
  procedure Inverse(aMatrix : Matrix);
  procedure InitForTest;
  end;
implementation
procedure SquareMatrix.InitForTest;
begin
  put(1,1,1); put( 2,1,2); put( 1,1,3); put(-1,1,4);
  put(3,2,1); put( 1,2,2); put(-1,2,3); put( 3,2,4);
  put(4,3,1); put( 2,3,2); put( 2,3,3); put( 1,3,4);
  put(1,4,1); put(-1,4,2); put(-1,4,3); put( 2,4,4);
  and; { SquareMatrix.InitForTest }
  constructor SquareMatrix.Init;
  begin
    Matrix.Init(l,l,n);
  end; { Init }
  procedure SquareMatrix.Inverse;
  var
    pivotUsed, result : boolean;
    size : integer;
    a : Matrix;
    pivotRow, pivotColumn } array [ 1..222 ] of integer;
    i, j, k, l : integer;
    temp, pivot, det : dataType;
  begin
    { initialize }
    result := true;
    size := getColumns;
    a.Init(size, (2 * size), 'a');
    { Postavlja prosiranu matricu }
    for l := 1 to size do begin
      for j := 1 to size do begin
        a.put(aMatrix.at(l,j), l, j);
        if i = j then a.put(1,i,(size + j))
          else a.put(0,l,(size + j));
      end; { for j }
    end; { for l }
    { Postavlja determinantu }
    det := 1.0;
    { Pocetak Gaus-Jordanove eliminacije }
    for k := 1 to size do begin
      { Bira maksimalni pivot }
      pivot := 0.0;
      for l := 1 to size do begin

```

```

for j := 1 to size do begin
  pivotUsed := false;
  if k <> 1 then
    for i := 1 to (k - 1) do
      if (i = pivotRow[i]) or
         (j = pivotColumn[j])
      then pivotUsed := true;
  if not pivotUsed then begin
    if abs(a.at(i,j)) > pivot then begin
      pivot := a.at(i,j);
      pivotRow[k] := i;
      pivotColumn[k] := j;
    end; | If abs(a1) |
  end; | If pivotUsed |
end; | for j |
end; | for i |
{ Ispituje singularnost matrice }
if pivot = 0 then begin
  result := false;
  WriteLn('Premature exit');
  exit;
end; | if pivot |
| Menja vrednost determinante |
det := det * pivot;
| Normalizuje red |
for j := 1 to (2 * size) do
  a.put( a.at(pivotRow[k], j) / pivot, pivotRow[k], j);
| Eliminiše preostale redove |
for i := 1 to size do begin
  temp := a.at(i, pivotColumn[k]);
  for j := 1 to (2 * size) do begin
    if i <> pivotRow[k] then begin
      a.put(a.at(i,j) - ( temp * a.at(pivotRow[k], j) ),
            i,j);
    end; | if i |
  end; | for j |
end; | for i |
end; | for k - glavna petlja |
| Uređivanje redova i ispravljanje znaka determinante |
for k := 1 to (size - 1) do begin
  if pivotRow[k] << pivotColumn[k] then begin
    det := - det;
    for j := 1 to (2 * size) do begin
      temp := a.at(pivotRow[k],j);
      a.put( a.at(pivotColumn[k],j), pivotRow[k],j);
      a.put(temp, pivotColumn[k],j);
    end; | for j |
    for j := (k + 1) to size do
      if pivotRow[j] = pivotColumn[k] then
        pivotRow[j] := pivotRow[k];
  end; | if pivotRow |
| Premesta stupce inverzne matrice |
for l := 1 to size do
  for j := 1 to size do
    put( a.at(l, (size + j)), l, j );
end; | for k |
end; | Inverse |
end; | SqMat |

```

Samo su tri nova metoda u klasi **SquareMatrix**. Prvi je **Init**. On poziva "gornji" metod **Matrix.Init**, i to sa istim prvim i drugim parametrom, tako da se dobije kvadratna matrica. Metod **InitForTest** neposredno postavlja jednu matricu radi testiranja. Konačno, glavni novitet u ovoj klasi je metod **Inverse**, koji izračunava inverznu matricu. Sam po sebi, ovaj metod spada u duže i zasniva se na Gauss-Jordanovoj eliminacionoj šemi.

Evo test programa, **mm1SqMat**, iz datoteke **MMTSOMAT.PAS**:

```

program mm1SqMat;
uses Crt, SqMat, Mat;
var
  a, b : SquareMatrix;
  c : Matrix;
begin;
  ClrScr;
  writeln('Program mm1SqMat, testiranje matricne inverzije');
  writeln;
  a.Init(4, 'a ');
  a.InitForTest;
  a.wr;
  writeln;
  b.Init(a.GetColumns, 'b ');
  b.inverse(a);
  b.wr;
  writeln;
  c.Init(a.GetRows, b.GetColumns, 'c = a - b ');

```

```

  c.mult(a, b);
  c.wr;
  a.done; b.done; c.done;
end. | mm1SqMat |

```

U programu **DSSQMAT.PAS** koristimo metod **InitForTest** da postavimo probnu matricu. Sa **Inverse** onda izračunavamo njenu inverznu matricu, koja je takođe objekat tipa **Matrix**. Proizvod probne (promenljiva **a**) i inverzne matrice (promenljiva **b**) je identična matrica (promenljiva **c**), kao što se vidi na slici 3.

Program mmSqMat, testiranje matricne inverzije

a	1.0000	2.0000	1.0000	-1.0000
	3.0000	1.0000	-1.0000	3.0000
	4.0000	2.0000	2.0000	1.0000
	1.0000	-1.0000	-1.0000	2.0000
b	2.5000	-1.2500	-0.2500	3.2500
	-1.0000	1.0000	0.0000	-2.0000
	-2.5000	0.7500	0.7500	-2.7500
	-3.0000	1.5000	0.5000	-3.5000
c = a * b	1.0000	-0.0000	-0.0000	0.0000
	0.0000	1.0000	0.0000	-0.0000
	0.0000	0.0000	1.0000	-0.0000
	0.0000	0.0000	0.0000	1.0000

Slika 3. Rezultat programa **dsSqMat**.

Promenljive tipa **Matrix** fragmentiraju dinamičku memoriju, što može uticati na potonje promenljive istog tipa. Zato ne bi trebalo naizmenično izvršavati metode **done** i **init** za različite promenljive. Rizik takve i sličnih grešaka je cena koju plaćamo za kreiranje matrica tokom izvršavanja programa a ne tokom prevođenja.

Vektorske klase – **ColumnMatrix** i **RowMatrix**

U klasi **dsSqMat** zapravo smo koristili vektore, tj. matrice sa samo jednim redom ili stupcem. Sada ćemo formalno definisati kao nove klase, **RowMatrix** i **ColumnMatrix**. Evo jedinica **mmRowMat** i **mmColMat** iz datoteke **MMROWMAT.PAS** i **MMCOLMAT.PAS**.

```

unit mmRowMat;
{ Definise jednostubacnu matricu }
interface
uses Mat;
type
  RowMatrix = object(Matrix)
  constructor Init(i : Integer; n : string);
  function at(i : integer) : dataType;
  procedure put(value : dataType; i : integer);
  end;
implementation
constructor RowMatrix.Init;
begin
  Matrix.Init(1, i, n);
end; { Init }
function RowMatrix.at;
begin
  at := Matrix.at(1,i);
end; { at }
procedure RowMatrix.put;
begin
  Matrix.put(value, 1, i);
end; { put }
end. | mmRowMat |

```

```

unit mmColMat;
{ Definise jednorodnu matricu }
interface
uses Mat;
type
  ColumnMatrix = object(Matrix)
  constructor Init(i : Integer; n : string);
  function at(i : integer) : dataType;
  procedure put(value : dataType; i : integer);
  end;

```

```

implementation
constructor ColumnMatrix.init;
begin
  Matrix.init(l, 1, n);
end; { init }
function ColumnMatrix.at;
begin
  at := Matrix.at(l,1);
end;
procedure ColumnMatrix.put;
begin
  Matrix.put(value, l, 1);
end; { put }
end. { mmColMat }

```

Klasa Point

Tačke su vrlo važni objekti. Potrebne su nam i u dvo- i u trodimenzionalnim prostorima. Jedina prava dilema je kako organizovati nasleđivanje. Tradicionalno razmišljanje vodi u formiranje apstraktne klase **Point**, koju bi nasleđile klase **Point2** za tačku sa dve dimenzije, odnosno, **Point3** za tri dimenzije. Sasvim je moguće da klasa **Point** (umesto klase **Point2**) bude jedina roditeljska klasa, a da je **Point3** nasleđuje direktno. To bi značilo da su sve trodimenzionalne tačke sadržane u dvodimenzionalnim tačkama. Jedan način bi bio da dvodimenzionalne tačke budu podskupovi trodimenzionalnih tačaka, tako da bi onda klasa **Point3** bila roditeljska. To zvuči dobro (=od opšteg ka posebnom), ali će onda dvodimenzionalne tačke uvek sadržati treću dimenziju, što povećava memorijске zahteve. Ovdje ćemo prikazati još jedan način organizovanja klase tačaka, biće to klasa **Point** iz jedinice **Pt (PT.PAS)**. Umesto dve posebne klase tačaka, imaćemo samo jednu, i to kao naslednik klase **RowMatrix**. Time izjednačavamo klase tačaka u dve i tri dimenzije i stičemo sposobnost da tačke zadajemo dinamički, tokom izvršenja programa (metod **init**).

Klasa **Point** je opšteprisutna u grafici pa sadrži veliki broj metoda. Jeste velika, ali se ta dužina ne treba uplašiti - nije nastala odjednom, nego sam je dopunjavao prema potrebama. Zapravo, unutar klase **Point** metodi se «bavi od sebe» dele u različite grupe. To je označeno komentarima, jer sintaksnih sredstava za grupisanje metoda unutar klase u Turbo Pascalu nema.

Uvek prvo pišemo metod **init**. Nema potrebe za novim metodom **done** u ovoj klasi, što ne znači da on ne postoji: njoga, naravno, nasleđujemo iz klase **RowMatrix**. Metod **init** koristi **RowMatrix.init** i zauzima više mesta nego što tačka ima koordinata. Poslednji deo tačke je nula, ali mora postojati jer homogena matrice linearnih transformacija (vldi definiciju kasnije u tekstu) zahtevaju dodatnu dimenziju u reprezentaciji tačke.

Posle metoda **init** bilo je prirodno dopisati operacije nad tačkama za koje iz matematike znamo da su česte i korisne. Takvi su sledeći metodi:

- dimension** - broj koordinata u tački,
- negated** - izmena znaka svih koordinata,
- scalarMult** - množenje svih koordinata brojem,
- pointMult** - množenje tačke tačkom, koordinatu po koordinatu,
- scalarPlus** - dodavanje broja (skalara) svakoj koordinati,
- pointPlus** - sabiranje tačaka, koordinatu po koordinatu,
- scalarSub** - oduzimanje broja od svake koordinata,
- pointMinus** - oduzimanje tačaka, koordinatu po koordinatu,
- scalarDivision** - deljenje koordinata brojem,
- dotProduct** - skalarni umnožak tačaka,
- translate** - transliranje tačke,
- abs** - apsolutna vrednost tačke, koordinatu po koordinatu,
- round** - zaokruživanje koordinata,
- trunc** - odsecanje koordinata,
- max** - najveća koordinata tačke,
- min** - najmanje koordinata tačke.

Dakle do prve grupe metoda došli smo na osnovu opštih matematičkih postavki. Da bismo mogli testirati napisano, uvodimo drugu grupu metoda koji olakšavaju praktičan rad. Kao prvo, uvodimo razne načine za neposredno postavljanje i očitavanje koordinata tačke. Metodi **putX**, **putY**, **putZ** postavljaju odgovarajuće koordinate tačke, dok ih funkcije **x**, **y**, **z** očitavaju. Ovi metodi su bili dovoljni da se napiše test program.

Kasnije su došli metodi **intersection** i **distance**, zbog odgovarajućih proračuna u klasi **Line** (linije). Ovi metodi pripadaju klasi **Point** samo po tome što je njihov rezultat (ili zato što su njihovi argumenti) - tačke. Konceptualno uzevši, oba metoda pripadaju klasi **Line**:

- intersection** je preseka linija, zadatih dvema parovirna tačaka.
- distance** je dužina između dve tačke. Ovak metod je mogao biti upisan u bilo koju drugu klasu.

Poslednja grupa metoda crta tačke. Osnovni metod se zove **draw**, ali je isto što i **show**: oba pozivaju lokalni metod **internalDraw** koji zapravo crta. Razlog za uvođenje dvaju metoda za istu stvar je što se obično crta sa **draw**, ali u priručniku za Turbo Pascal se za to koristi reč **show**.

Metod **hide** je suprotno od **draw** i **show** - briše tačku sa ekrana.

Metodi **initGraph** i **closeGraph** samo pozivaju istoimene naredbe iz jedinice **Graph**. Zašto ih onda uvodimo i u klasu **Point**? Prosto, zgodno je

slati ih kao poruke tačkama, jer bi inače u glavnom programu morali da pozivamo i jedinicu **Graph**.

Promenljiva **visible** pamti stanje vidljivosti za tačku. Njene vrednosti su logičke, **false** i **true**, i postavljaju ih metodi **draw** i **hide**. Metod **isVisible** vraća tekuću vrednost promenljive **visible**.

Mnogi metodi u klasi **Point**, npr. **abs** i **round**, zovu se isto kao i naredbe samog Turbo Pascala iz jedinice **System**. «Naivno» navođenje ovih imena bez naznake jedinice vodi sintaksnim greškama. Zato ih unutar metoda klase **Point** preclizamo prefiksom **System**, i pišemo **System.abs**, **System.trunc** i sl.

Konačno, metod **perspTrans** je napisan tek pošto je osnovni kod klase **Point** postao stabilan; **perspTrans** izgleda zasađa van konteksta, ali će razlog za njegovo ubacivanje naknadno postati jasan.

Možemo zaključiti da organizovanje metoda u klasu nije linearan proces. Skoro da je nemoguće tek tako sestri, zamisliti i ispisati baš sve što će budući korisnici klase zahtevati.

Klasa **Point**, iz jedinice **mmPt** i datoteke **MMPT.PAS**, izgleda ovako:

```

unit mmPt;
{ Definiše tačku }
interface
uses Mat, mmColMat, mmRowMat, mmSqMat, Graph, mmPrdPri;
type
  mat33 = SquareMatrix;
  pointPtr = ^Point;
  Point = object(RowMatrix)
    visible : boolean;
    procedure perspTrans(pp : PerspectiveProjection; point3D : Point);
    constructor Init(i : Integer; n : string);
    function dimension : Integer;
    procedure negated;
    procedure scalarMult(value : dataType);
    function pointMult(p : Point) : dataType;
    procedure scalarPlus(value : dataType);
    procedure pointPlus(p : Point);
    procedure scalarSub(value : dataType);
    procedure pointMinus(p : Point);
    procedure scalarDivision(value : dataType);
    procedure abs;
    function dotProduct(p : Point) : dataType;
    procedure translate(p : Point);
    procedure round;
    procedure trunc;
    procedure sphericalCoordinates(wd, psi, eps : dataType);
    function max : dataType;
    function min : dataType;
    function x : dataType;
    function y : dataType;
    function z : dataType;
    procedure putX(value : dataType);
    procedure putY(value : dataType);
    procedure putZ(value : dataType);
    procedure intersection(a1, a2, b1, b2 : Point);
    function distance(p : Point) : dataType;
    procedure InitGraph;
    procedure closeGraph;
    function isVisible : boolean;
    procedure draw;
    procedure show;
    procedure hide;
    procedure internalDraw;
    procedure moveTo(p : Point);
  end;
implementation
constructor Point.init;
begin
  visible := false;
  RowMatrix.init(l + 1, n);
  RowMatrix.put(0, l + 1);
end; { Init }
procedure Point.negated;
var i : integer;
begin
  for i := 1 to dimension do
    put(-at(i), i);
  end; { negated }
function Point.dimension;
begin
  dimension := RowMatrix.getColumns - 1;
end; { dimension }
procedure Point.scalarMult;
var i : integer;
begin
  for i := 1 to dimension do
    put(at(i) * value, i);

```

```

end; { scalarMult }
function Point.pointMult:
var l : Integer;
value : dataType;
begin
value := 0;
for i := 1 to dimension do
value := value + at(i) * p.at(i);
pointMult := value;
end; { pointMult }
procedure Point.scalarPlus;
var l : Integer;
begin
for l := 1 to dimension do
put( at(l) + value, l);
end; { scalarPlus }
procedure Point.pointPlus;
begin
translate(p);
end;
procedure Point.scalarSub;
begin
scalarPlus(-value);
end; { scalarSub }
procedure Point.pointMinus;
var l : Integer;
begin
for l := 1 to dimension do
put( at(l) - p.at(l), l);
end; { pointMinus }
procedure Point.scalarDivision;
begin
scalarMult(1 / value);
end; { scalarDivision }
procedure Point.abs;
var l : Integer;
begin
for l := 1 to dimension do
put( System.abs(at(l)), l);
end; { abs }
function Point.dotProduct:
begin
dotProduct := pointMult(p);
end; { dotProduct }
function Point.max;
var l : Integer;
v : dataType;
begin
v := at(1);
for l := 1 to dimension do
if v < at(l) then v := at(l);
max := v;
end; { max }
function Point.min;
var l : Integer;
v : dataType;
begin
v := at(1);
for l := 1 to dimension do
if v > at(l) then v := at(l);
min := v;
end; { min }
procedure Point.translate;
var l : Integer;
begin
for i := 1 to dimension do
put(at(i) + p.at(i), i);
end; { translate }
procedure Point.round;
var l : Integer;
begin
for i := 1 to dimension do
put( System.round(at(i)), i);
end; { round }
procedure Point.trunc;
var l : Integer;
begin
for i := 1 to dimension do
put( System.trunc(at(i)), i);
end; { trunc }
procedure Point.sphericalCoordinates;
begin
put( cos(psi) * sin(eps), 1);
put( cos(eps), 2);
put( sin(psi), 3);

```

```

scalarMult(wd);
end; { Point.sphericalCoordinates }
function Point.x;
begin
x := at(1);
end; { x }
function Point.y;
begin
y := at(2);
end; { y }
function Point.z;
begin
z := at(3);
end; { z }
procedure Point.putX;
begin
if dimension > 0 then
put(value, 1);
end; { putX }
procedure Point.putY;
begin
if dimension > 1 then
put(value, 2);
end; { putY }
procedure Point.putZ(value : dataType);
begin
if dimension > 2 then
put(value, 3);
end; { putZ }
procedure Point.intersection;
var
k1, k2, n1, n2 : dataType;
InvSm, sm : SquareMatrix;
constants : ColumnMatrix;
results : RowMatrix;
begin
k1 := (a2.y - a1.y) / (a2.x - a1.x);
k2 := (b2.y - b1.y) / (b2.x - b1.x);
n1 := - a1.x * k1;
n2 := - b1.x * k2;
InvSm.Inlt(2, 'InvSm');
with sm do begin
inlt(2, 'sm');
put(k1, 1, 1); put(-1, 1, 2);
put(k2, 2, 1); put(-1, 2, 2);
InvSm.Inverse(sm);
end; { with sm }
with constants do begin
inlt(2, 'constants'); put(-n1, 1); put(-n2, 2);
end; { with constants }
results.inlt(2, 'results');
results.mult(InvSm, constants);
put(results.at(1), 1);
put(results.at(2), 2);
sm.done; InvSm.done; results.done; constants.done;
end; { intersection }
function Point.distance;
var s : dataType;
l : Integer;
begin
s := 0;
for i := 1 to dimension do
s := s + sqr(at(i) - p.at(i));
distance := sqrt(s);
end; { distance }
procedure Point.inltGraph;
var gd, gm : Integer;
begin
gd := Graph.Detect;
Graph.inltGraph(gd, gm, '');
if Graph.graphResult <> Graph.grOk then halt(1);
end; { InltDraw }
procedure Point.closeGraph;
begin
Graph.closeGraph;
end; { closeGraph }
function Point.isVisible;
begin
isVisible := visible;
end; { isVisible }
procedure Point.show;
begin
visible := true;
Graph.putPixel(System.trunc(x), System.trunc(y), getColor);
end; { show }

```

```

procedure Point.hide;
begin
  visible := false;
  putPixel(System.trunc(x), System.trunc(y), getBkColor);
end; { hide }
procedure Point.draw;
begin
  show;
end; { draw }
procedure Point.internalDraw;
begin
  show;
end; { internalDraw }
procedure Point.moveTo;
var i : integer;
begin
  hide;
  for i := 1 to dimension do
    put(p.at(i), i);
  show;
end; { moveTo }
procedure Point.perspTrans;
var tx, ty, tz, e1, e2, e3 : dataType;
begin
  with pp do begin
    tx := point3D.x + o1;
    ty := point3D.y + o2;
    tz := point3D.z + o3;
    e1 := d1*tx + d2*ty + d3*tz;
    e2 := d4*tx + d5*ty + d6*tz;
    e3 := d7*tx + d8*ty + d9*tz;
    put(xfactor * xdistance * (e1 / e2), 1);
    put(xfactor * xdistance * (e3 / e2), 2);
  end; { with pp }
end; { perspTrans }
end. { mmPt }

```

Test program, `mmPt`, nije dugačak, ali rezultira u čak tri ekrana. Prva dva, 4 i 5, su rezultati osnovnih poruka objektima klase `Point`. Treći ekran na prikazujemo, jer jedna jedina tačka nacrtana na ekranu izgleda skoro isto kao i prazan ekran.

Program `mmdsPt` – testiranje klase `Point`

a	1.0000	-3.0000	4.0000	0.0000
a.negated	-1.0000	3.0000	-4.0000	0.0000
a.scalarMult	-3.0000	9.0000	-12.0000	0.0000
b	1.0000	1.0000	1.0000	0.0000
pointMult(b) =		-6.0000		
a.scalarPlus(b)	-1.0000	11.0000	-10.0000	0.0000
a.pointPlus(b)	0.0000	12.0000	-9.0000	0.0000
a.scalarSub(3)	-3.0000	9.0000	-12.0000	0.0000
a.pointMinus(b)	-4.0000	8.0000	-13.0000	0.0000
a.scalarDivision(10)	-0.4000	0.8000	-1.3000	0.0000
a.abs	0.4000	0.8000	1.3000	0.0000
dotProduct(b) =		2.5000		

Slika 4. Prvi ekran programa `mmdsPt`.

Poruka `Init(3,")` definiše trodimenzionu tačku, koja ■ štampa kao četvorka brojeva. Slično, poruka `Init(2,")` proizvodi trojku brojeva, koji predstavljaju tačku u ravni.

Slika 6 takođe prikazuje izračunati presek dvaju linija.

Program `mmdsPt` – drugi ekran

```

a.max 1.3000
a.min 0.4000
a.x 10.0000
a.y 2.6000
a.z 30.0000

```

a.translate(b)	11.0000	3.6000	31.0000	0.0000
a.round	11.0000	4.0000	31.0000	0.0000
a.trunc	13.0000	4.0000	31.0000	0.0000
sphericalCoordinates(1,pi,pi)	0.0000	-1.0000	0.0000	0.0000
p1	1.0000	1.0000	0.0000	0.0000
p2	0.0000	0.0000	0.0000	0.0000
p3	0.0000	1.0000	0.0000	0.0000
r, presecna tacka linija	1.0000	0.0000	0.0000	

Slika 5. Drugi ekran programa `mmdsPt`.

```

program mmDsPt; { Demo za Point }
uses Crt, mmPt, TCUtil, Graph;
var a, b, r, p1, p2, p3, p4 : Point;
    result : Mat.dDataType;
    g : word;
    i : integer;
begin
  ClrScr;
  writeln('Program mmDsPt – testiranje klase Point'); writeln;
  with a do begin
    Init(3, 'a ');
    put(1,1); put(-3,2); put(4,3); wr;
    negated: changeName('a.negated'); wr;
    scalarMult(3,0); changeName('a.scalarMult '); wr;
    b.Init(a.dimension, 'b ');
    b.put(1,1); b.put(1,2); b.put(1,3); b.wr;
    result := pointMult(b);
    writeln('pointMult(b) = ', result:13:4);
    scalarPlus(2); changeName('a.scalarPlus(b)'); wr;
    pointPlus(b); changeName('a.pointPlus(b)'); wr;
    scalarSub(3); changeName('a.scalarSub(3)'); wr;
    pointMinus(b); changeName('a.pointMinus(b)'); wr;
    scalarDivision(10);
    changeName('a.scalarDivision(10)'); wr;
    abs: changeName('a.abs'); wr;
    writeln('dotProduct(b) = ', dotProduct(b):13:4);
    g := GetKey;
    ClrScr;
    writeln('Program mmDsPt – drugi ekran');
    writeln('a.max', max:13:4);
    writeln('a.min', min:13:4);
    putX(10); putY(2.60); putZ(30);
    writeln('a.x', x:13:4);
    writeln('a.y', y:13:4);
    writeln('a.z', z:13:4);
    translate(b); changeName('a.translate(b) '); wr;
    round: changeName('a.round'); wr;
    putX(13.4);
    trunc: changeName('a.trunc'); wr;
    sphericalCoordinates(1, pi, pi);
    changeName('sphericalCoordinates(1,pi,pi)'); wr;
  end; { with a }
  r.Init(2, 'r, presecna tacka linija ');
  p1.Init(2, 'p1'); p2.Init(2, 'p2');
  p3.Init(2, 'p3'); p4.Init(2, 'p4');
  p1.putX(1); p1.putY(1);
  p2.putX(0); p2.putY(0);
  p3.putX(0); p3.putY(1);
  p4.putX(1); p4.putY(1);
  p1.wr; p2.wr; p3.wr; p4.wr;
  r.intersection(p1, p2, p3, p4);
  r.wr;
  g := getkey;
  a.InitGraph;
  OutText('Program mmDsPt – ekran br 3, grafka');
  p1.putX(100); p1.putY(100); p1.draw;
  g := getkey;
  p2.scalarPlus(200);
  p1.moveTo(p2);
  g := getkey;
  a.closeGraph;
  a.done; b.done; r.done; p1.done; p2.done; p3.done; p4.done
end. { mmDsPt }

```

Linearne transformacije

Do sada izložene klase odnosile su se na opšte vrste matrica koje imaju primenu u raznim tehničkim, matematičkim i statističkim aplikacijama. Ovdje se bavimo samo primenama matrica u računarskoj grafici. Unapređenja su uglavnom u klasi **SquareMatrix**, što znači da ćemo uglavnom konstruisati njene potklase.

Matricama na efikasan način predstavljamo linearne transformacije. One su opšteg oblika

$$y = Ax$$

gde su x i y vektorske promenljive, A matrice. Geometrijski uzet, x i y su tačke u ravni. U računarskoj grafici je uobičajeno predstavljati tačke kao vektore koji su za po jednu dimenziju veći nego što je dimenzija prostora kome tačke pripadaju. Tako se dvodimenzionalna tačka pretvara u vektor 1×3 , a matrica transformacije koja odgovara takvoj tački je tipa 3×3 . Najopštiji oblik takve matrice je

$$A = \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ e & f & 1 \end{bmatrix}$$

Treći stubac je fiksiran na jedinični vektor, pa izgleda kao i da ne učestvuje u transformacijama. Postoji samo da bi matrica A bila kvadratna, pa da kasnije možemo da joj izračunavamo inverznu matricu (ako postoji).

Matrica u kojima je poslednji stubac kao u navedenoj su homogena. Definišemo novu klasu, **HomogeneousMatrix**, kao potklasu za **SquareMatrix**. Jedina razlika je u metodi **init**. Sada on mora da predefiniše strukturu poslednjeg stubca:

```
unit mmHomMat;
{ Definise homogenu matricu }
interface
uses SqMat;
type
  HomogeneousMatrix = object(SquareMatrix)
  constructor Init(rows : Integer; n : string);
  end;
Implementation
constructor HomogeneousMatrix.Init;
var l : Integer;
begin
  SquareMatrix.Init(rows,n);
  zero;
  for i := 1 to getColumns do
    put(0, getColumns, i);
  put(1, getColumns, getColumns);
end; { init }
end. { mmHomMat }
```

Translaciju definišemo kao

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ Tx & Ty & 1 \end{bmatrix}$$

Direktnim proračunom dobijamo jednačine translacije u ravni:

$$X = x + Tx$$

$$Y = y + Ty$$

$$1 = 1$$

Rotacija oko koordinatnog početka za ugao θ definiše se kao:

$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

dok se homotetija definiše kao

$$S = \begin{bmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Ove tri matrice zajedno mogu predstaviti bilo koju transformaciju tačke u ravni. Zahvaljujući nasleđivanju, ove matrice je lako napisati kao potklase za **HomogeneousMatrix**. Za translaciju definišemo klasu **TranslationMatrix**, u datoteci **MMTRNMAT.PAS**.

```
unit mmTrnMat;
{ Definise matricu translacije }
interface
uses mmHomMat, mmPt;
type
  TranslationMatrix = object(HomogeneousMatrix)
  constructor Init(point : Point; n : string);
  procedure Inverse(hm : TranslationMatrix); virtual;
  end;
```

```
implementation
constructor TranslationMatrix.Init;
var l, size : Integer;
begin
  size := 1 + point.dimension;
  HomogeneousMatrix.Init(size,n);
  identity;
  for i := 1 to (size - 1) do
    put(point.at(i), size, i);
  end; { init }
procedure TranslationMatrix.Inverse;
var t, i : Integer;
begin
  t := hm.getRows;
  for l := 1 to t - 1 do begin
    put(-hm.at(t,i), t, l);
    end; { for i }
  end; { inverse }
end. { mmTrnMat }
```

Matricu translaciju zgodno definišemo tačkom, tj. objektom tipa **Point**. Program **ttTrnMat** ilustruje kako se translacija koristi.

```
program ttTrnMat;
{ Testiranje translacione matrice }
uses Crt, mmTrnMat, mmPt;
var
  a, c : TranslationMatrix;
  b, d : Point;
begin
  ClrScr;
  writeln('Program ttTrnMat, testiranje translacije');
  writeln;
  b.Init(2, 'b, tacka u ravni');
  b.put(3,1);
  b.put(5,2);
  b.wr;
  a.Init(b, 'a, matrica translacije ');
  c.Init(b, 'c, inverzna za a ');
  a.wr;
  c.Inverse(a);
  c.wr;
  d.Init(b.dimension, '');
  d := b;
  d.changeName('d, promenjen znak');
  d.negated;
  d.wr;
  a.done; b.done; c.done; d.done;
end. { ttTrnMat }
```

Program ttTrnMat, testiranje translacije

b, tacka u ravni			
	3.0000	5.0000	0.0000
a, matrica translacije			
	1.0000	0.0000	0.0000
	0.0000	1.0000	0.0000
	3.0000	5.0000	1.0000
c, inverzna za a			
	1.0000	0.0000	0.0000
	0.0000	1.0000	0.0000
	-3.0000	-5.0000	1.0000
d, promenjen znak			
	-3.0000	-5.0000	0.0000

Slika 6. Rezultat programa **ttTrnMat**.

Rezultat programa **ttTrnMat** vide se na slici 6.

Promenljiva b je tipa **Point**, tj. tačka dimenzije 2. Njome postavljamo matrice a i c . Obe su matrice translacije, c je a inverzna za a . Promenljiva d je tačka sa izmenjenim predznakom. Pošto imamo metod **negated** za tačke, možemo inverziju translacione matrice pisati i jednostavnije, pozivom metoda **negated** kao argumenta za **init**.

Nastavak u sledećem broju

Zaključaj podatke javnim ključem

DRAGAN PLESKONJIĆ, dipl.ing.

Ovaj tekst predstavlja logički nastavak teksta iz prošlog broja *Mog mikro* pod naslovom »Zaštita tajnih podataka i programa«. Članak se bavi problemom šifrovanja podataka odnosno poruka koji se šalju na daljinu nepoverljivim kanalom. Nepoverljivim kanalima možemo danas praktično smatrati sve osim ličnog prenosa poruke. Razrađuje se pristup prenosu podataka korištenjem šifrovanja prema tzv. kriptosistemsima sa javnim ključevima (public key systems).

Zamislite da trebate rešiti sledeći problem:

Morate poslati kofer u kom su dragocene stvari (ili poruka, podaci, programi), svom prijatelju (prijateljići) koji živi u udaljenom gradu. Niko ne sme znati sadržaj imate na raspolaganju kurira u prenosni pul kome previše ne verujete. Naime, ako kurir sazna šta je u koferu (poruci) svima će izbrbljati. Vi to, naravno, ne smete dozvoliti. Šta ćete učiniti? Možete zaključati kofer. Ali kako da ga primalac otključa kad je ključ kod vas? Ne smete ključ poslati istim kurirom jer mu ne verujete. Znači, treba vam i drugi kurir koji će nositi ključ. A šta ako se kurir usput nađu i pogledaju kofer? Ne dolazi u obzir, a još je i skupo.

Šta je rešenje? Možda:

1. Kurir prvo odnese kofer (razume se, zaključan), preda ga, a onda se vrati po ključ.

2. Zaključate kofer, kurir ga odnese. Primalac zaključa kofer svojim katancom. Onda kurir vrati kofer vama. Otključate svoj katanac (kofer je sada zaključan katancom primaoca). Kurir odnese kofer primaocu, on ga otključa i uzme ono što je za njega.

Malo komplikovano, zar ne? I skupo istovremeno.

A šta ako postoji brava koja se jednim ključem zaključava, a drugim otključava? Neobičajeno, ključ koji je bravu zaključao, ne može je i otključati. Ta dva ključa su inverzna. Dakle, na bravu primenite transformaciju T_1 koja je iz stanja otključano (S_0) prevodi u stanje zaključano (S_1). Ovo se obavlja uz pomoć ključa K_1 . Da biste bravu otključali koristite transformaciju T_2 uz pomoć ključa K_2 . Svaki od ključeva (K_1, K_2) poništava transformaciju koju izvede onaj drugi. Dakle važi (u pojednostavljenoj zapisu):

$T_2 \circ T_1$ tj. transformacije T_1 i T_2 su međusobno inverzne

$K_1 * K_2 = 1$ tj. ključevi K_1 i K_2 su međusobno inverzni (znak * ne označava pristo množenje)

$T_1(S_0, K_1) = S_1$ tj. T_1 prevodi poruku iz stanja S_0 u stanje S_1 uz pomoć ključa K_1

$T_2(S_1, K_2) = S_0$ tj. T_2 prevodi poruku iz stanja S_1 u stanje S_0 uz pomoć ključa K_2 .

Uzastopna primena transformacija T_1 i T_2 daje identitet, tj. ostavlja poruku u stanju u kom je prethodno bila. Dakle, onaj ko želi da mu se dragocena poruka šalje i ostanu tajne kod prenosa, treba proizvesti dva inverzna ključa. Jedan od ta dva ključa pošalje onima koji mu poruke šalju i objavi:

»Sve što mi šaljete zaključajte (šifrujte) ovim ključem (K_1).«

Niko, naravno, ne može otključati (dešifrovati) poruku, jer niko nema ključ koji za to služi (K_2). To može učiniti samo onaj ko je generisao par ključeva K_1 i K_2 , pod uslovom da se ne nađe neka oštroumna glava koja će na osnovu javnog ključa K_1 izračunati i tajni K_2 (tj. inverzni). Dakle, tajni ključ mora biti jednosmerna funkcija javnog ključa, tj. ne sme se moći, u raspoloživom vremenu i raspoloživim sredstvima, izračunati tajni ključ na osnovu poznatog javnog. Tada se ne može izvršiti ni transformacija šifrovanje (zaključavanje) poruke u dešifrovanje (otvorenje), tj. šifrovani tekst je takođe jednosmerna funkcija otvorenog teksta.

Proširimo ovaj problem:

Racimo da postoje dvije osobe, A i B, koje žele međusobno slati poruke. Nisu baš poverljivi prema drugima. Žele biti sigurni u sledeće:

1. Onaj ko šalje poruku želi biti siguran da niko drugi neće pročitati poruku osim onoga kome je ona namenjena.

2. Onaj ko prima poruku želi biti siguran od koga ona stvarno potiče, tj. želi biti siguran da mu možda neko ne šalje poruku u tuđe ime (lažno predstavljanje).

Šta učiniti? Rešenje je u sledećem:

Osobe A i B generišu po par inverznih ključeva. Tako osoba A generiše dva inverzna ključa K_A i K_A' , pri čemu je K_A javni, a K_A' tajni ključ. Na sličan način osoba B generiše ključeve K_B i K_B' .

Sada A i B javno objave ključeve K_A i K_B , dok K_A' i K_B' čuvaju u tajnosti. Pretpostavimo da niko nije u stanju iz javnih ključeva izvesti tajne (kasnije će biti objašnjeno kako se takvi ključevi stvarno generišu i koriste). Kako osoba A šalje poruku osobi B? Prvo poruku zaključa, tj. transformiše korištenjem svog tajnog ključa K_A' (postupak ove transformacije biće takođe objašnjen kasnije). Zatim tako zaključanu poruku zaključa još jednom, i to javnim ključem osobe B, tj. sa K_B koji je svima poznat. I takvu poruku može bez brige poslati na bilo koji način osobi B. Niko je neće moći dešifrovati osim osoba B.

Kako osoba B dešifruje primljenu poruku? Vrio jednostavno. Prvo primeni javni ključ osobe A, tj. K_A . Time »poništa« efekat koji je osoba A proizvela svojim tajnim ključem K_A' (ovo može učiniti bilo ko). Uspesšan ishod ove operacije potvrđuje osobi B da je stvarni izvor poruke osoba A, jer samo ona (osoba A) zna

svoj tajni ključ. Poruka još nije dešifrovana. Sledeći korak koji preduzima osoba B je transformacija uz pomoć ključa K_B' , tj. svog tajnog ključa (ovo može učiniti jedino osoba B jer jedino ona zna svoj tajni ključ). Time je osoba B došla do otvorenog (dešifrovanog) oblika poruke koju joj je poslala osoba A.

Na analogan način osoba B može slati poruku osobi A. Ovo se može proširiti na još više osoba koje žele na sličan način komunicirati. Znači, svaka osoba koja želi učestvovati, određuje par ključeva (svoj javni i svoj tajni ključ). Javni ključ objavi i zahteva da svi koji mu šalju poruke obave šifrovanje uz pomoć njenog javnog ključa. Ukoliko još želi i proveru verodostojnosti (autentičnosti) poruke, onda zahteva da onaj koji šalje poruku vrše šifrovanje svojim tajnim ključem, kao što je objašnjeno.

Ovakvi sistemi se u kriptografiji nazivaju kriptosistemi sa javnim ključevima (public key systems). Problemi koje treba rešiti su sledeći:

- određivanje para ključeva (javnog i tajnog) tako da se iz javnog ključa ne može izvesti tajni

- određivanje transformacije (funkcije preslikavanja) poruke (otvorenog teksta, programa, podataka) u šifrovani tako da se ne može izvršiti inverzija bez poznavanja tajnog ključa.

Znači, tajni ključ je jednosmerna funkcija javnog ključa i šifrovani tekst je jednosmerna funkcija otvorenog teksta. Mehanizam koji ove probleme rešava, i za sebe ima dosta složen matematički aparat koji će ovde biti kratko prikazan. Cilj je da se omogući razumevanje postupka određivanja ključeva prema zahtevima koji se postavljaju i da se objasni princip šifrovanja i dešifrovanja.

Šta je to jednosmerna funkcija

Jednosmerne funkcije imaju sledeću karakteristiku: za datu vrednost X lako je izračunati vrednost funkcije $Y=F(X)$, ali ako je dato $Y=F(X)$, onda ne postoji lak put za izračunavanje X . Drugim rečima, ne postoji izvodljiv računski metod za određivanje $F^{-1}(Y)$. Za kontinualne i analitičke funkcije lako je izvršiti numeričku inverziju. Zato se kao jednosmerne funkcije koriste diskontinualne i proizvoljne funkcije.

Pojam jednosmeran nije apsolutan, već zavisi od toga za koliko mnogo izračunavanja kažemo da je »nemoguće velik broj«. Kaže se da je, za datu vrednost funkcije $F(X)$, izračunavanje vrednosti za X »računski nemoguć« ako ovo izračunavanje zahteva onoliko vremena i sredstava koliko se ne može izdvojiti. Takođe se zna da sa vrlo srećnim pogađanjem ili blaznim vrednosti funkcije $F(X)$ kojima

odgovara poznato X , ovo izračunavanje postaje nepotrebno.

Jedan sistem sa javnim ključem napravljen je tako da koristi očevito težak problem računanja logaritama u aritmetici po modulu q (q je prost broj i polje $GF(q)$ ima q elemenata $\{0, 1, \dots, q-1\}$; GF je oznaka za Galo-ovo polje od q elemenata).

Neka je:

$$Y = a^X \text{ mod } q \quad 1 \leq X \leq q-1 \quad (1)$$

Ovde se sva računanja obavljaju u aritmetici po modulu q (npr. $5^2 \text{ mod } 11 = 25 \text{ mod } 11 = 4$). Pri tome je u gornjoj formuli a fiksan primarni element polja $GF(q)$ (gde su stepeni broja a nenula elementi $1, 2, \dots, q-1$ od $GF(q)$). Tada se X računa kao logaritam od Y u bazi a preko $GF(q)$.

$$X = \log_a Y \text{ preko } GF(q) \text{ za } 1 \leq Y \leq q-1 \quad (2)$$

Računanje Y iz X je lako i zahteva najviše $2 * \log_2 q$ množenja. Na primer:

$$a^{16} = ((a^8)^2)^2 = a^8 * a^8 \quad (3)$$

Računanje X iz Y je mnogo teže i za određene pažljivo izabrane vrednosti q zahteva operacija reda $q^{1/2}$.

Svaki korisnik generira jedan nezavisan slučajni broj Z izabran iz intervala $\{1, 2, \dots, q-1\}$. On drži X_i u tajnosti, ali $Y_i = a^{X_i} \text{ mod } q$ (4)b

i objavljuje ga kao javni ključ. Kada korisnici i i j žele komunicirati tajno, oni kao svoj ključ koriste:

$$K_{ij} = a^{X_i X_j} \text{ mod } q \quad (5)$$

Korisnik i računa K_i iz Y_i na sledeći način:

$$K_i = Y_i^{X_i} \text{ mod } q \quad (6)$$

$$= (a^{X_i})^{X_i} \text{ mod } q \quad (7)$$

$$= a^{X_i^2} \text{ mod } q = a^{X_i^2} \text{ mod } q \quad (8)$$

Korisnik j dobija K_j na jednostavan način:

$$K_j = Y_j^{X_j} \text{ mod } q \quad (9)$$

Drugi korisnik može izračunati K_{ij} iz Y_i i Y_j , na primer, računanjem:

$$K_{ij} = Y_i^{(a^{X_j})^{Y_j}} \text{ mod } q \quad (10)$$

Dakle, ako su logaritmi preko $GF(q)$ (u aritmetici po modulu q) lako izračunljivi, sistem može biti provaljen. Ako nema načina da se izračuna K_{ij} iz Y_i i Y_j , a da se prvo ne dobije X_i ili X_j , sistem je siguran.

Ako je q prosti broj mnogo manji od 2^n , sve veličine se mogu predstaviti kao b -bitni brojevi. Stepenovanje onda zahteva najviše $2b$ množenja u aritmetici po modulu q , dok logaritmiranje zahteva $q^{1/2} = 2^{b/2}$ operacija, koristeći najbolji poznati

algoritam. Ako $b=200$, potrebno je najviše 400 množenja da se izračuna Y , iz X , ili K_1 , iz Y , i X , iz Y i X , međutim, izračunavanje K_1 , iz Y , i Y , zahteva najmanje 2^{100} ili oko 10^{30} operacija.

Opis RSA algoritma sa javnim ključem

Eksponencijalna funkcija poslužila je na poseban način Rivestu, Shamiru i Adlemanu da stvore (RSA) kriptosistem sa javnim ključem. Oni su koristili činjenicu da je nalazjenje velikih (npr. 100-cifrenih) prostih brojeva računarski dosta lako, ali faktorizacija proizvoda dva takva broja je, čini se, računarski praktično nemoguća.

Opisaćemo ukratko postupak određivanja ključeva i šifrovanja po ovom algoritmu.

Korisnik A bira dva vrlo velika prasta broja, P i Q , i njihovim množenjem dobija broj N . Broj N je javan, ali njegovi faktori P i Q drže se u tajnosti. Koristićanjim P i Q korisnik A može izračunati funkciju $\phi(N)$ (što je broj prirodnih brojeva manjih od N i relativno prostih sa N) kao:

$$\phi(N) = (P-1) * (Q-1) \quad (11)$$

On onda bira drugi broj E iz intervala od 1 do $\phi(N)-1$. Ovaj broj je takođe javan. Poruka je prikazana kao niz brojeva M_1, M_2, \dots gde je svaki broj između 0 i $N-1$. Šifriranje se provodi na svakom bloku M korišćenjem javnih informacija E (to je javni ključ) i N (modulo, tj. aritmetika u kojoj se radi) kao:

$$C = M^E \text{ mod } N \quad (12)$$

gde C predstavlja šifrovani blok. Korišćenjem tajnog broja $\phi(N)$ korisnik A može lako izračunati broj D (tajni ključ) tako da:

$$(E * D) \text{ mod } \phi(N) = 1 \quad (13)$$

(ekvivalentno $E * D = k * \phi(N) + 1$).

Ovim D obezbeđena inverznost ključeva E i D i time i inverznost postupaka šifrovanja i dešifrovanja. Ako E ima zajednički faktor sa $\phi(N)$, onda D ne postoji i mora se izabrati drugi E . Onda zbog:

$$X^{k\phi(N)+1} = X \text{ mod } N \quad (14)$$

za sve cele brojeve između 0 i $N-1$ i k dešifrovanje je lako izvodljivo potenciranjem C na D -tu potenciju:

$$C^D = M^{ED} = M^{k\phi(N)+1} = M \text{ mod } N \quad (15)$$

Primer: Neka su izabrani $P = 17$ i $Q = 31$. Tada je $N = PQ = 527$ i $\phi(N) = (P-1)(Q-1) = 480$. Ako je $E = 7$, onda je $D = 343$ ($7 * 343 = 2401 = 5 * 480 + 1$). Ako je $M = 2$, onda

$$C = M^E \text{ mod } N \\ = 2^7 \text{ mod } 527 \\ = 128$$

Za šifrovanje bio je potreban javni ključ. Za dešifrovanje neophodan tajni ključ:

$$M = C^D \text{ mod } N \\ = 128^{343} \text{ mod } 527 \\ = 128^{265} * 128^{10} * 128^4 * 128^2 * 128^1 \text{ mod } 527 \\ = 35 * 256 * 35 * 101 * 47 * 128 \text{ mod } 527 \\ = 2 \text{ mod } 527$$

Dakle, postupci šifrovanja i dešifrovanja su isti, stim što se u prvom slučaju koristi javni ključ (E) nad otvorenim tekstom (M), a u drugom tajni ključ (D) nad šifratom (C). Radi se sa numeričkim interpretacijama teksta (ASCII ili drugim).

Ilustracija šifrovanja po RSA algoritmu

Glavni problem kod realizacije šifrovanja po ovom algoritmu je izračunavanje jednosmerne funkcije koja je oblika:

$$m^i \text{ (mod } n).$$

Ovde se koristi jedan dosta efikasan algoritam koji omogućuje izračunavanje eksponencijalne funkcije oblika m^i ponavljanjem kvadriranja i množenja u sledećih nekoliko koraka:

1. korak: Neka je h_1, h_2, \dots, h_k binarna reprezentacija broja i
2. korak: $c := 1$
3. korak: $i := k$
4. korak: $c := c^2 \text{ mod } n$
5. korak: ako je $h_k = 1$, onda $c := cm \text{ mod } n$
6. korak: $i := i - 1$
7. korak: ako $i < 0$, onda kraj, inače idi na korak 4.

Procedura za izračunavanje jednosmerne funkcije mogla bi izgledati ovako (programski jezik pascal):

```

Procedura JednosmFunkcija (Poruka, Ključ, Modulo : integer, Var Sifra : Integer);
{ Procedura dobija kao ulaz: numeričku interpretaciju bloka poruke, ključ i modulo. Izlaz je šifrovani blok. Poruka, Ključ, Modulo i Sifra su globalne varijable tipa Integer. }
Var A : array [1..500] of 0..1; {max. 500 binarnih cifara za predstavljanje ključa}
I, J : integer;
Begin
  { Konverzija ključa u binarni }
  I := 0;
  While Ključ > 0 do
  begin
    I := I + 1; A[I] := Ključ mod 2;
    Ključ := Ključ div 2;
  end;

  { Šifrovanje jedinica teksta predstavljene numerički u varijabli Poruka }

  Sifra := 1;
  For J := 1 downto 1 do
  begin
    Sifra := Sifra * Sifra;
    If A[J] = 1 then Sifra := Sifra * Poruka;
    Sifra := Sifra mod Modulo;
  end;
End; { JednosmFunkcija }

```

Naravno, elegantnije rešenje je izdvojiti konverziju ključa u binarni oblik u posebnu proceduru. Konverziju treba obaviti samo jednom na početku šifrovanja/dešifrovanja poruke (šifrata). Rezultat se može smestiti u globalnu varijablu, tj. polja koje sadrži binarne cifre ključa. U ovom primeru $A[0]$ je binarna cifra najmanje težina.

Primer šifrovanja. Rečimo da je trebalo po ovom metodu šifrovati sledeći tekst:

"SNAGA RSA ALGORITMA JE U PROBLEMU FAKTORIZACIJE VELIKIH BROJEVA"

Uzmimo, za ilustraciju, proste brojeve:

$$P = 9 \mid Q = 11$$

Znači, radi se o aritmetici po modulu

$$N = P * Q = 99$$

Potrebno je izračunati funkciju

$$\phi(N) = (P-1) * (Q-1) = 80$$

Neka javni ključ bude $E = 3$ i tajni ključ $D = 27$, jer je

$$(3 * 27) \text{ mod } 80 = 1$$

Broj decimalnih cifara	Broj potrebnih operacija	Potrebno vreme
50	$1,4 * 10$	3,9 sati
75	$9,0 * 10$	104 dana
100	$2,3 * 10$	74 godine
200	$1,2 * 10$	$3,8 * 10$ godina
300	$1,5 * 10$	$4,0 * 10$ godina
500	$1,3 * 10$	$4,2 * 10$ godina

Kao rezultat dobija se sledeći šifrat (numerička interpretacija):

28 71 01 46 01 45 90 28 01 45 01 45 46 09 90 36 80 19 01 45
 10 26 45 54 45 37 90 09 08 45 26 19 54 45 18 01 44 80 09 90
 36 53 01 27 36 10 26 45 55 26 45 36 44 36 17 45 90 09 10
 26 55 01

Napomena: U ovom primeru slova su interpretirana tako da je numerička interpretacija $A = 1, B = 2$ itd. Šifrovani su blokovi od po jednog znaka (slova) i korišteni su mali prosti brojevi. To je učinjeno radi jednostavnije ilustracije. Ovakvo šifrovanje tekst lako bi se, naravno, dešifrovalo. Naime, u ovom primeru sistem je degradiran na prostu zamenu znakova.

Primena RSA algoritma

U stvarnim primenama RSA algoritma uzimaju se veliki prosti brojevi za generisanje ključa (recimo brojevi od oko 50 - 100 dekadskih cifara). Kao jedan blok poruke tada se uzima čitav niz od po 20 do 30 znakova. Recimo, u ASCII kodu, interpretacija niza znakova "ABCDEFGH" bi bila "6566576869707172". Kod ovakvog šifrovanja ne postoji mogućnost faktorizacije broja N , niti izračunavanja tajnog ključa na bazi javnog (što se takođe svodi na faktorizaciju). Dakle, stvarna sigurnost RSA algoritma počiva na nemoci današnjih računera i algoritama da u raspoloživom vremenu izvrše faktorizaciju velikih brojeva, odnosno izračunaju inverz jednosmerne funkcije.

Prikažimo ovde neke podatke o vremenima potrebnim za faktorizaciju velikih brojeva:

RSA algoritam se smatra veoma sigurnim. Omogućava šifrovanje komuniciranja velikog broja učesnika, pri čemu se može obezbediti mogućnost provere identiteta izvora poruke, ukoliko je potrebno. Sistem sa javnim ključevima je novi koncept u kriptografiji, s obzirom na rešenje veoma osetljivog problema distribucije ključa.

Probleme, kod primene ovog algoritma, predstavljaju: dosta složen postupak šifrovanja, relativno niska brzina šifrovanja i složen postupak određivanja parova ključeva (javni i tajni). Naravno, ovi problemi se mogu ublažiti kvalitetnim algoritmi i brzim softverom i hardverom.

RSA algoritam je naročito pogodan za primenu kada više učesnika komunicira. Vrlo pogodno se može primeniti kod banaka, gde veliki broj poslovnica komunicira sa centralom ili kod sličnih ustanova kod kojih je bitna tajnost poruka (transakcija), a poslovna mreža je razgranata.

Ovo pismo neću da započnem sa ustaljenim »Poštovani...«. Posudili su rečenicu g. A. iz uvoznika u broju iz novembra meseca 1990. g. (kojem ne dajem za pravo): »Nećete dugo da izlazite ako budete ovako poslovni.« Usput – poštedite nas viceva!

Pretplatnik »Mog mikra« sam od dana kada se časopis prvi puta pojavio. Na žalost, u poslednje vreme, samo zbog toga jer ga već drugu godinu dobijam nesplatno (Zabavni matematički zadaci). Kažu da se poklonjenom konju ne gleda u zuba, pa već duže vremena oklevam da napišem ovih nekoliko reči. Na kraju, odlučio sam da vam ih napišem iako se bojim da će sve proći kao »lejanje na mesec.«

Osnovna stvar zbog koje mislim da se »Moj mikro« definitivno pokvario su reklame! Po mojoj gruboj oceni na oko osamdeset strana ne baš kvalitetnog (da ne kažem lošeg) papira, reklame su na preko četrdeset. Po meni (pa na samoj po meni), to je apsolutno previše. Šteta je stalna koja moraju pasti za listova na kojima su otšampane same reklame, a tu i tamo je poneki čitljiv tekst. Poslednji tekst koji sam čekao sa veseljem bile su pomenute matematičke zagadke. Na zato što sam, nakon toga, dobio besplatno »Moj mikro,« jer se za njega, kakav je u poslednje vreme, uistinu ne vredni truditi. Nakon toga započeo je sinuosi ples po sistemu: »Sad ga vidiš, sad ga ne vidiš« bez ikakvog objašnjenja. Za takvu gimnastiku, cena od 35 dinara je prokleta visoka. Neću da navodim primere stranih stručnih revija, kvalitetnijih i jeftinijih od »Mog mikra« pošto ih sami dobro poznajete! Što se cene tiče, samo mi nemojte zapovesti staru pesmicu u troškovima štampa-

nja. Ili možda ne dobijate od reklama i oglasa toliko novca da bi »Moj mikro« mogao biti deljan besplatno, ako bi ga neko hteo uzeti – ne da ga kupi.

Meli oglašali su, kako izgleda, bili druga žalost »Mog mikra«. Kako možete u uvodniku tumačiti veselu vest o usvajanju Zakona o autorskim pravima, a na jednoj od narednih strana objavljujete piratske male oglase neke čak na srpskohrvatskom u slovenskom izdanju!). Pretnja sudom vam, bar što se mene tiče, ne može oprati fleks najgorog licemerija, čemu je, bez sumnje, kumovao novac.

Mnogo reči ne treba gubiti ni na kvalitet opisa igara, ni tome sam već ranije pisao. Obzirom na kvalitet, igre ne zaslužuju ni da se pojave na stranama »Mog mikra«. Molim vas da ne objavljujete opise igara koje nemaju bar ocenu osam, iako bi i kod drugih, prije objavljivanja, trebalo uvesti malo cenzure. Kao primer, mogu da navedem opis trećeg dela Sierrinog Larryja, objavljenog u novemberskom broju 1990. Opis je veoma slab, mnogo mu nedostaje da bi bio potpun. G. Jezidžić bi morao igru završiti pre nego što je u ruke uzeo čovku i papir. Možda bi mu to uspelo da je imao pred sobom legalnu verziju programa uz koju se dobije interesantna tanka sveska bez koje nam igra bar dva puta stane, da ne možemo dalje. Prvi puta, kada skupimo približno petsto poena...

Bar tu budite kvalitetniji, inače, stvarno ćete prestati izlaziti!

Jernej Čop
Mesesnelova 10
Ljubljana

Gospodine Pavlović, vaš vid je potpuno ispravan. U međuvremenu

i ja sam dobio novu verziju Micro-softovog paketa Works V2.0 od 29. 6. 1990. Odmah sam počeo testirati paket i upoznavati se sa novostima u njemu. Uistinu, nije moguće promeniti tipku SKIP MACRO. U mojoj ranijoj verziji V1. 05, koja ima MSKEY.COM, to je bilo moguće upotrebom MSKEY.COM s jednostavnim dodatkom (SET SKIP kada tipke).

Kada su integrirali makronaredbe u paket, na tu olakšicu su zaboravili. U ovoj verziji je i rad sa makronaredbama dosta teži nego u verziji 1.05. Naravno, ne može se sve imati. Zbog toga su dodali nekoliko sitnica poput kalendara, kalkulatora itd. Tako nam je od dve mogućnosti koje sam predložio u prošlom pismu, ostala samo jedna. Za znak »ž« tipku je potrebno dva puta pritisnuti.

Što se tiče paketa WORKS 2.0, slažem se sa vama da se radi o integriranom paketu koji u budućnosti obećava. WORKS 2.0 zadovoljava sve potrebe sekretarica i daktilografkinja, jer je obrada teksta jednostavnija za korištenje nego u velikim paketima (Word za MS-DOS, Word-Perfect 5.1, WordStar 8.0 itd.), ima mnoge od njihovih mogućnosti, brži je kod označavanja strana i nekih drugih akcija od, na primer, Word-Perfecta 5.1. Pored toga, tabelarno računanje (spreadsheet) je isto zasnovano kao čuveni Lotus 1-2-3, ima ugrađen prikaz dijagrama koji se aktivno menja. Baza podataka je lepo uređena, kao i komunikacioni deo. Sve je lepo upakovano, svi meniji i načini rada su lepo dizajnirani. Podatke je moguće kombinovati. Dodan je ispo pripremljen i poučan deo za vežbe i upoznavanje paketa. Senzitivna pomoć (help) je lepo uređena, da ne govorim u dobrom priručniku.

Ukratko, dobar paket iz dobre programske kuće

Na kraju, još nešto: na disketu (oznaka SETUP), nalaze se dve datoteke README.WPS, na kojim su komentari kojih nema u priručniku, kao i COUNTRY.WPS u kojoj mogu samo da kažem da, kada sam je pročitao, srce mi je zaigralo, lica mi je ozario smešak. U njoj je zapisaano kako možemo da »nacionalizujemo« naš paket. U našem WORKS imeniku je datoteka INTL.RSC (jednostavno rečeno international Resource File), u koju se mogu ugraditi parametri za jugoslovenske konvencije. Mogu se ugraditi simboli za novčanu jedinicu, oblik datuma, decimalni zarez, tačke, imena meseci u godini, abecedni poredak itd. Nakon kratkog pregleda te datoteke mislim da bi se isplatio taj paket prilagoditi našim uslovima i to bi trebalo da uradi službeni predstavnik Microsofta u Jugoslaviji.

Do tada, dok uistinu ne zaživi tržište programske opreme, moraću sam da prilagođavam INTL.RSC datoteku svojim potrebama.

Andrej Ivanušić
Gospodstvska 43
Maribor

Ispravka

U broju 12/1990. na str. 7, objavljena je vest pod naslovom Apple se otreznilo. U njoj je navedeno da najjeftiniji macintosh classic u nas staje 10.000 din. Tada je stvarna cena bila 15.500 a sada je 20.460 din (bez poreza na promet) Na želju čitalaca, evo i adrese Appleovog zastupnika za Jugoslaviju: ACS (Adria Computer Systems), d. o. o., Maksimirska 111, 41000 Zagreb, tel. (041) 232-114, faks (041) 232-128

ZABAVNI MATEMATIČNI ZADACI

REŠENJA ZADATAKA IZ DECEMBARSKOG BROJA

SOLITER

Arhitekt Avbelj ima kancelariju na osimnom spratu, advokat Boh na trećem spratu, lekar Mum na petom spratu i ekonomist Starc na petnaestom spratu.

KANDIDATI

Sozdesetposam kandidata je slušala fiziku i hemiju

NOVINAR

Pošto nema porodica bez dece, svaka porodica mora imati bar jednu kćerku koja je jedina dete ili je sestra eventualnih dečaka. Dakle, mora biti bar toliko

devojčica, koliko je porodica. Pošto je više dečaka nego devojčica, ukupno dece mora biti više od odraslih osoba. Jer su u svakoj porodici samo dva odrasla osobe. Međutim, to je u suprotnosti s prvom činjenicom u izveštaju.

DUGAČAK BROJ

Jedno od rešenja je sledeće: 123456789100
Između brojeva smo, dakle, postavili decimalnu zapetu.

NOVI ZADACI

PROFESOROV TELEFONSKI BROJ

Redoznali student pitao je profesora razveseljena kako mu kod njegova osokupna raseljanosti uspeva da zapamti svoj telefonski broj. Odgovor je bio sledeći:

»Savim jednostavno! Pamtim samo činjenicu da je moj telefonski broj jedini sedmocifreni broj koji je, napisan obrnutim redom brojeva, delitelj samog sebe!«

Utvrdila profesorov telefonski broj! **TAJANSTVENA KOČKA**



Na slici vidimo tri različita pogleda na istu tajanstvenu kocku. Utvrdite koliko je tačaka naslikanih na stranici koja se nalazi nasuprot šestice na prvom skicu!

NEOBİČAN BROJ

Između brojeva 2 i 200.000.000.000.000 nalazi se samo jedan broj koji je istovremeno kvadrat, kub i peti stepen odgovarajućih prirodnih brojeva. Pokažite taj broj!

JANEZOVE NEČAKINJE

Janez i profesor Blisk vodili su interesantan razgovor

Janez: »Verovatno će vam izgledati interesantno da su moja starost i starost moje tri nećakinje prosti brojevi. Suma svih iznosi 50.«

»U tom slučaju odmah mogu da odredim starost tvojih nećakinja,« odgovorio je profesor koji je znao koliko godina ima Janez.

Pokušajte i vi, bez obzira što vam je starost Janeza nepoznata, da utvrdite starost Janezovih nećakinja! (Vodite računa da broj 1 nije prost broj!)

NAGRADE

Rešavaocima je veliku glavobolju izazvao zadatak SOLITER, pa zato reševaljamo da dočijena »rešenja« još jednom proverile, da ne budu u suprotnosti sa pretpostavkama i činjenicama datičnog zadatka. Mnogi prestaju da rešavaju kad dobiju prividno rešenje, ne znajući da su kod zaključivanja napravili neku grešku. Zato treba naučiti i određenu kritičnost prema svom rešavanju.

Ze iscrpna rešenja smo jednogodišnjom pretplatom nagradili Marina Bajca, Fani Grumove 14, Zalog, 61260 Ljubljana – Polje

Drugi nagradeni su:

Astrid Barthi, Goleška 15, 38000 Priština;

Dunja Ignjatović, Vinogradski vrtac 15, 11138 Beograd;

Jane Gabec, Arja vas 16, 83001 Petrovče;

Milanka Santelj, Čabranska 6, 61380 Cerknica.

Rešenja bar tri zadatka pošaljite do 1. MARTA 1991. na adresu: Revija Moj mikro, Tiltova 35, 61000 Ljubljana (Zabavni matematički zadaci). Nagrade su uobičajene: jednogodišnja preplata na reviju Moj mikro za najoriginalnija rešenja sva četiri zadatka i računarske nagrade za srećne koji se obrađuju izvlačenjem, sa bar tri pravilna rešenja (kasete, diskete, knjige).

STIMMGABEL

Tel.: 9943-4242/24-722
Faks: 9943-4242/210044
A-9500 VILLACH, Lederergasse 6

ATARI TT 030-4 +
MONITOR PTC-1426

29.000,- ATS neto

ATARI TT 030-4 +
MONITOR TTM 194/19'

35.000,- ATS neto

Skidz

• sportska simulacija • amiga, ST
• Gremlin • 9/9

TOMAŽ DVORAK

To je jedina prava simulacija »rolkanja« i vožnja biciklom BMX, napravljena za vašu vernu prijateljicu. Na odličnom uvodnom ekranu birate, da li ćete upotrebljavati »rolku« ili BMX, a zatim jedan od sedam poligona: Street – ulica, Park – park, Building Site – gradilište, China Town – kineski kvart, Canal – pristanište, Race – trka.

Slede još opcije SHOP I RACE. U trgovini možete da kupite (ako imate novac) različita pomoćna sredstva i dodatnu opremu za vašu »rolku« ili BMX. Preporučujemo vam da napre-



kupite bocu sa vodom (WATER BOTTLE), jer vam održava energiju. Kad kliknete na opciju RACE (trka) naći ćete se na vrlo dobro izrađenoj stazi BMX, gde se takmičite sa još jednim takmičarom kojim upravlja računar. Kod trke se vaša energija ne smanjuje. Savetujem vam da vozite odmah iza računarski upravljanih igrača i da ga preteknete tek na kraju.

A sada su na redu poligoni. Možete da ih birate redom, računar vas seli drugi poligon tek tada kad apsolvirate prvog. Cilj igre je da se pređu svi poligoni, a na svakom morate da skupite 75 odsto smeća (papiri, kore, konzerve) koja treba strpati u sanduke. Igrače vodite igračkom palicom. Dugmetom ubrzavate, a pomeranjem gore, dole, levo, desno, skrećete. Energiju za vreme igre obnavljate time što skupljate jabuke, limune, grožđe i drugo voće. Zatim je tu još novac: skupljate novčiće koji se vrte (vreda jedan dolar) i izgužvane novčanice koje vrede šest dolara. Novac možete da skupljate takođe atraktivnim skokovima preko skakaonice. Vozite najvećom brzinom i ispred skakaonice pritisnite FIRE Bicikl (rolka) će se podići i biće u vazduhu. Tada okrenite palicu za 360 stepeni i dobićete pet dolara. To ponovite više puta, jer je veoma važno da imate mnogo novca.

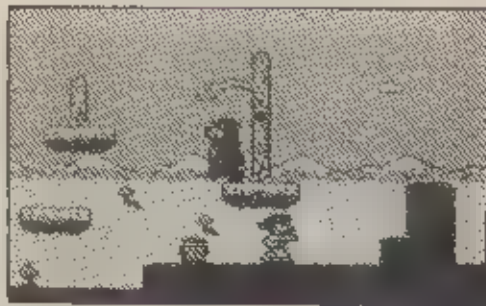
Na kraju još nekoliko saveta. Vozite što brže i ne ovrćite se na svako smeće. Pazite takođe na energiju, jer vam je na kraju svakog stepena maksimalno potrebna.

Yogy's Great Escape

• arkadna igra • spectrum, C 64, CPC • Hi-Tec 8/8

MITJA MLADKOVIČ

Sigurno ste već gledali crtiče sa simpatičnim medvedom Yogyjem. Sada ga drugi put susrećemo i u računaru. Programeri su se potrudili i napravili igru sa prilično dobrom grafikom i animacijom. Razočaranje je muzika koja se čuje samo na početku nivoa.



Ekran je podeljen na dva dela. U donjem, manjem nalaze se broj poena, vreme koje je potrebno za završetak nivoa i broj preostalih života (u početku samo tri). U gornjem delu se odvija igra.

1. nivo, JELLYSTONE PARK: Imate 2 minuta i 40 sekundi vremena. Jure vas veoma napadački raspoloženi skauti (?) koji vas ponekad gađaju i velikim balvanima, a i prilično manje opasne zmije. Cilj je da se pokupi što više automobilića. Kod svakog prikupljenog automobilića popunjava se sličica na dnu ekrana.

2. THE FOREST: Sada imate vremena 2 min. In 25 sek. Neprijatelji su isti. Međutim, skauti su postali mnogo opasniji, jer su se naoružali lovačkim puškama. Sada skupljate »delove« i svog (Yogyjevog) lica. Razlike između prvog i drugog nivoa skoro i nema.

3. THE WILD WEST: Na Divljem zapadu prillike su se za vas značajno pogoršale. Skauta više nema, ali su tu loše (divlje) raspoloženi Indijanci koji svojim lukovima i strelama postaju veoma opasni, pa ih zato izbegavajte. Vremena imate 2 minuta i 10 sekundi. Morate da pokupite sve džakove s novcem.

4. MUMBO JUMBO MARSH: To je ubedljivo najbolje napravljeni nivo, ali i najteži. Vremena imate 1 minut i 50 sekundi. Nestaju svi stari neprijatelji, ali se zato pojavljuju gomila novih. Najopasniji su duhovi (njihov plašuci usklik je odlično napravljen), tu su takođe pauzi, krvoločne ptice... dakle, sve što se može pronaći u dobroj močvari. Sada bar ništa ne treba da skupljate.

5. THE FAIRGROUND: Bez obzira što na ovom nivou imate vremena 1 minuta i 40 sekundi, nisam uspeo da stignem do kraja. Opet se pojavljuju skauti, zmije, tekuća traka – gomila neprijatelja.

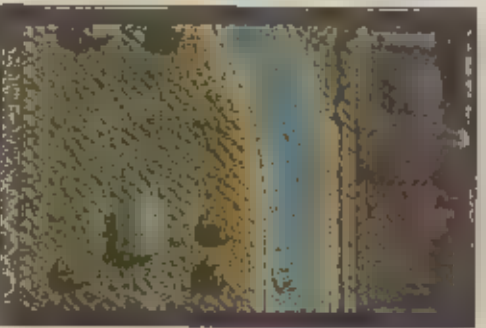
Igra će vas, uprkos dobroj grafici, teško zadržati pred računarom više od dva časa. Jer je tako teška da brzo možete da izgubite nerve i bacite kasetu i igračku palicu kroz prozor!

The Spy Who Loved Me

• arkadna igra • amiga, spectrum, C 64, CPC, ST • Tengen/Domark • 8/8

SAŠO ŠMALC

Saradnja Tengena i Domarka skoro uvek nam donosi dobru igru. Ovog puta su, verovatno, ustali na pogrešnu nogu. Njihova poslednja igra



je prosečan nastavak The Living Daylights.

James i njegova prijateljica imaju zadatak da uhvate šefa mafija u Londonu. U igri je osam stepeni. Vreme nije ograničeno, što je veliki plus. Za vreme vožnje možeš da odrađuješ brzinu. Ovdje važi stara poslovica: polako se daleko stiže. Za vreme vožnje skupljaj bonuse i izbegavaj ljude na ulici. Ubrzo će za tobom doći razbojnici. Smanji brzinu, povuci se na levu ivicu ulice i pošalji im rafal metaka u leđa. Kad se kolovoz račva moraš da se opredeliš za onaj pojas koji je pravi, jer ćeš inače nastradati. Moraš da paziš takođe, da prelethš mostove pravom brzinom. Drugi stepen je vožnja čamcem. Tu važe slična pravila kao na prvom stepenu. Oba stepena se smanjuju, a time raste takođe broj neprijatelja i prepreka na ulici ili u vodi.

Na kraju sledi čestitka i obavljeni posao. Grafika još nekako zadovoljava ali je ređa nezanimljiva.

Ninja Remix

• borilačka avantura • C 64, spectrum, ST, amiga • System 3 • 8/9

JASENKO KREJIĆ

Čim je stigla, ova igra je od naših »jedinih pravih izvora« dobila epitet »najboljeg ostvarenja Systema 3« i »hita meseca i 90. godine«. Međutim, stvarnost to sasvim pobija. Igra je u potpunosti kopija Last Ninja II sa nekim izmenama i prepravkama. Svi nivoi (Central Park, The Streets...) su potpuno isti kao u Last Ninja II, a osim sprajtova i grafike čak je i loading



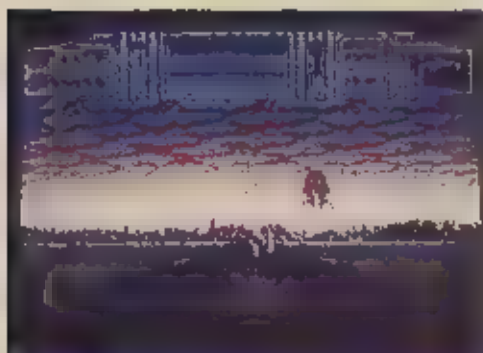
screen prije svakog nivoa ostao isti. Nekoliko pohvala bi se moglo izreći u ime muzike, koja je ili potpuno nova, ili miksovana muzika iz obadva prethodna dijela Last Ninje. Tu je i odlična i zanimljiva uvodna sekvencsa sa odličnim zvukom, grafikom i animacijom, ali koja postaje dosadna i oduzima vrijeme kad je 2-3 puta pogledate. Opšti je utisak da je System 3 izbacio Remix kako bi pojačao groznicu pre dolaska Last Ninje III. Sumnjam da će iko kupiti Remix sada kada je Last Ninja III već izašao.

Fire and Forget II: The Death Convoy

• arkadna igra • amiga, spectrum, C 64, CPC, ST, PC • Titus • 8/8

SERGEJ HVALA

Zemlja, dvadeset minuta u budućnost. Nuklearni rat je opustošio nekada rasvetalu planetu. Ljudi su se našli u retkim preostalim gradovima.



ili su odabrali drugi put – krstare po već upotrebljivim putevima sa još upotrebljivim vozilima, gde važi zakon snažnijeg. Instinkt za samoočuvanje nadvladao je razum. Uništavanje sopstvene rase postalo je, pored preživljenja, jedini cilj drumskih hordi. Na nesreću »građana« jedina horda je u napuštenom vojnom uporištu otkrila kamion ■ nuklearnom raketom snage od oko sto megatona. Kamion vozi prema najvećem gradu sa ciljem ■ ga nuklearnom eksplozijom potpuno uništi. Kamion i njegovu pratnju možete da uništite samo vi, svojim najmodernijim automobilom. Kao hrabri (i glupi) junak sedate ■ auto i krećete za konvojem. Da li će vam uspeti?

No, da li će vam uspeti da uz ovaj scenario ne pomislite na nesnalažljivost njegovog autora? Već je često bilo utvrđeno da programerima nedostaju ideje, a sada sam i ja u to potpuno ubeđen. Napraviti nastavak igre koja ima hiljadu puta upotrebljenu ideju, za to je dovoljan dokaz. Fire & Forget 2 je ■ dlaku slična Road Blastersima i Overlanderu, ali je od njih, ipak, nešto bolja. Grafika za amigu nije ništa naročito, zvuk takođe, a svetla tačka je samo animacija. Tačku na «-» dodaje činjenica što je igra obrada ■ igračkog automata.

Kao i kod pomenute dve igre, svoj auto vidite od pozadi. Naoružani ste bacačem loše nacrtanih vatrenih kugli (ili užarenih olovnih kugli ogromnog kalibra) koje lansirate pritiskom na vatra, ■ igračkoj palici i raketama koje najbliži cilj pretvaraju u prah i pepeo (HELP). U gornjem delu ekrana su stanje bodova (milionski brojač u stilu Space Harriera), količina raketa (u početku pet), preostali životi (kružići ispod silnice

automobila) i ekran na kome se ispisuju različiti podaci, na primer koliko je udaljeno čelo konvoja, odnosno kamion ■ raketom – kad se pojavi poruka »Visual Contact In« (broj), morate znati da ćete morati, kad brojač postigne nulu, da se sukobite s letećim čuvarom kamiona – da li možete da uzletite (Take Off), oznaka za kraj stepena itd. Na putu ka kamionu srećete različita vozila, od motocikla do statičkih samohodki i avione koji miruju na putu (!), a sve treba uništiti (ako uništite sve odjednom, dobijate bonus). Kad kamion bude uništen, možete ići na sledeći stepen. Ako izgubite sve živote, u daljnji će odjeknuti nuklearna eksplozija, pa će umesto grada ostati samo pustoš.

Varžuju za amigu možete da igrate takođe kursorским blokom tipki i razmakom, a na glavnom ekranu razmakom možete da odredite broj kredita (Credits: 0-3).

Kod igre možemo da pohvalimo sledeće: interesantna špica, dobra animacija, dinamičnost i dobro nacrtane slike uz poruke Game Over i Hall Of Fame, a možemo da joj zamerimo očigledno istrošenu ideju, slabu grafiku i zvuk. U Konvoju smrti zabavu će pronaći pristalice Road Blastersa.

Double Dribble Basket

● sportska simulacija ● amiga, C 64
● Kónami ● 8/8

ANDREJ BOHINC

Bez obzira što je TV Sports Basketball nadmašuje na svim područjima. DDB je jedna od boljih košarkaških simulacija u poslednje vreme. Ističe se, pre svega, detaljima koji obogaćuju igru. U početku umesto šifre elegantno pritisnite RETURN. Sledi klasičan izbor trajanja četvrtine (5-30 min), ekipo i stepena težine.

U sledećoj sceni vidite raku gledalaca koja se silva u dvoranu. Pre nego što pukne od zauzetosti, pritisnite SPACE.

Igra počinje pozivom sudije »JUMP OFF«! Svi igrači (po pet sa svake strane) su jednako visoki, a razlikuju se samo prema broju dresa. Rasuti su po celom igralištu, kao kad bi igrali fudbal! Kad

imate loptu ■ rukama, možete da je dodate onom igraču koji treperi. Protivniku je oduzimate ako u njegovoj blizini pritisnete pucanje.

Ne isplati se napade završavati bacanjem iz daleka ili sa poludistancije, jer se s lakoćom može doći pod koš. Zabijanja su efektivno prikazana i povećana, mada uvek ne uspevaju.

Igri možemo da zamerimo što ne poznaje lične greške, zamene, tajm-aut. Rezultat, vreme igranja i napada u gornjem delu ekrana slabo se vide. Autori su veću pažnju posvetili sporednim efektima. Pripremili su pravi NBA show, kakav mogu da organizuju samo Amerikanci. Gledaoci skaču sa svojih sedišta i navijaju za svoju ekipu.



Događaje na igralištu prate podsticajne himne i šou u pauzi. Tu su i maskote ekipa koje smetaju sudijama kod prekida. Ako pobedite, očekuje vas lepa završna scena.

DDB će vam dosaditi već posle nekoliko partija. Ili ćete je zavoleti zauvek.

Murder

● avantura ● amiga, C 64 ● Ocean ● 8/8

SAŠO ŠMALC

Ocean je poznat prema igrama sa otrcanom idejom. Ovog puta je izdao nešto zaista sveže. Amaterski detektiv je pozvan na mesto zločina. Njegov je zadatak da za dva časa reši problem ubistva, pre dolaska Savezne policije. U početku ekran prekriva prva strana lista Daily Chronicle.

COMPUTER DIVISION

- prodaja licencirane programске opreme
- prodaja CD-ROM
- savetovanje kod kupovine SW ja HW
- usluge u području DTP
- odkupujemo in livozimo VAŠE PROGRAME
- tražite sjenovnik 1

Tel: 061 316-343

TRIAS WTC PREDSTAVLJA



DALMATINOVA 4, LJUBLJANA

ART DIVISION

Elektronski seminar

UVOD U ART DESIGN & DTP

- pregled DTP paketa, grafičkih alata
- OCR programi
- VENTURA, COREL DRAW, Bitstream
- tehnologija izrade logo, internog glasila..
- Trajanje: 5 dana

Tel: 061 310-660

EXPORT/IMPORT DIVISION

- za vas uvozimo po najpovoljnijim cehatna robu koju tražite
- posudite svoju robu za izvoz

Tel: 061 310-833

FILE - možete učitati i snimiti podešavanja formule ili Instant replay.

OPTIONS - možete podešavati osjetljivost miša i količinu grafičkih detalja za vrijeme trke. Što je više grafičkih detalja, formula se sporije kreće, ali to je toliko neznatno da je detalje najbolje poslušati na high.

Grafika, animacija i muzika su fenomenalni i po mišljenju mnogih premašuju Test Drive II.

Za korištenje instant replaya potrebno je 1 Mb memorije. Ako u toku vožnje želite pozvati Instant replay, pritisnite ESC i izaberite opciju instant replay. Kada se nakon toga želite vratiti u trku izaberite opciju resume driving.

Final Countdown

• arkadna avantura • amiga, ST
• Demonware • 9/9

SERGEJ HVALA

Sunčani sistem. Lokacija: okolina Plutona. Zemaljsko vreme: 30. jul 2437. Vasiona stanica Telcom 1 u zemljinoj orbiti je na ivici sunačnog sistema primetila veliki asteroid, registrovan pod kodom AST2437:07-01. Glavni računar je protiv tajanstvenog asteroida najpre ispalio predajnik, a on se posle dva dana učutio. Izračunata verovatnoca kvara: 47.8%. Zatim su bila lansirana dva istraživačka robota-sonde koje su istovremeno prestale da rade prilikom dodira sa asteroidom u orbiti Jupitera. Verovatnoca isto-



vremenog kvara obe sonde: 0%. Dan kasnije: asteroid je napredovao do kružnice Marsa. Na površini AST2437:07-01 primaćena je energetska aktivnost. Verovatnoca povezivanja istovremenim prestankom delovanja robota: 98.7%. Glavni računar na stanici Telcom 1 odlučio je da na asteroid pošalje čoveka, da ga pretraži i da ga, ako zaključa da je bilo kako opasan za stanicu ili Zemlju, uništi poslavljanjem mina na prava mesta. Do trenutka kad će se asteroid sresti sa Zemljom, preostalo je još 12 dana i jedan čas, a to je takođe ukupno vreme odmereno za mislju.

Žene u uniformama 2437. godine nisu ništa posebno. Jedna od njih ste takođe vi, pripadnica grupe za diverzije (popularno: Zelene baretke) koje provodi još jedan dan na Telcomu 1, uz vašlju hranu, veštačku tečnost i zaslepljujuću svetlost neonskih svetala. Kad vas, dakle, odaberu kao pravu za ovaj zadatak, malo će vam laknuti.

Dok se svojim vasionim vozilom približavate asteroidu, najpre ćete primetiti bleštanje iz unutrašnjosti. Kad dođete u neposrednu blizinu, pripremajući se za ateriranje, iz kratera ispod vas projurice zaslepljujući zrak. Samo svom nujevitom refleksu možete da zahvatite što ste se na vreme lansirali. Vaše vozilo nestaje u vatrenoj kugli. Još pre nego što se onesvestite, uspevate da raketni ranač usmerite prema najbližem krateru. Zatim vas obavlja tama.

Takav je scenario za novu arkadnu avanturu u stilu legendarnog Obligatora, zvanu Poslednje odbrojavanje. Sada ste promene rad u ulozi

lepe plavuše (dakle, niste Sigourney Weaver) koji mora, kao što se vidi iz priče, tačno za 1040400 sekundi da obavi poslednji zadatak i da pobjegne sa AST2437:07-01. Asteroid je, kako otkriva posle buđenja, u stvari gigantska maskirana vasiona krstarica, naseljena inteligentnim bićima. Zadatak neće biti lak, jer je put do izlaza posejan energetskim zidovima, laserima, zaključanim vratima, a bića prilikom dodira sa vama odmah aktiviraju alarm koji treba isključiti najkasnije u pola minuta. Jer je inače - »krc«! Da sve ne bi bilo tako crno, na leđima imate raketni ranač (jet-pac), na ručl posebnu narukvicu čiji se podaci (preostalo vreme, status: zelen - sve u redu, žut - alarm uključen, crven - odbrojavanje, količina goriva u jet-pacu (fuel) i broj mina) projiciraju na gornji deo igračkog ekrana, a u glavi biste misli i obilna hrabrost. Tu su još liftovi koji vas prenose sa jednog nivoa na drugi (gore + hitac), zatim računarski terminali, iz kojih možete dobiti korisne podatke o strukturi krstarice, o lokaciji njenih najosetljivijih delova, izlazu, minama itd. Na tlu su razbacani najrazličitiji predmeti, od klopk koje mogu odmah da vas ubiju, do kapsula sa gorivom (E) za vaš jet-pac koji je, inače, veoma žedan. Predmete skupljate sa dole - hitac. Kad izgubite život, brojač vremena se povećava za sto, a transporter vas postavlja na polazište. Za odbranu nemate ništa, ni pesnice, a sve zavisi samo od količine goriva u raketnom ranaču.

Taktika za uspešno igranje je takva kao u Obligatoru, Danu Dareu i sličnim igrama - napredujete sporo i oprezno, ne rizikujete bez potrebe, a pre svega štedite gorivo. Samo tako ćete u odmerenom vremenu pobedi sa krstarice, pre nego što je unište visokoenergetska polja mina.

Ako Final Countdown uspoređujemo sa drugim igrama, u takvom verzijom, videćemo da spada u sam vrh, a možda čak i na prsto. Grafika i animacija su bolje nego u Obligatoru, a zvuk je nešto slabiji. Možda treba dodati još neko oružje da bi se mogli zadovolje i oni koji mogu bez krvi (crvene ili zelene). No, to je stvar ukusa. Igra zauzima dve diskete, a ima i mali nedostatak - vlasnici neproširenih amiga moraće da se odreknu zvučnih efekata. I sve spoljne diskete jedinice moraće da isključe.

The Killing Game Show

• arkadna igra • amiga, ST • Psygnosis
• 9/9

MARKO SEKULIĆ

Psygnosis je još jednom izbacio na tržište odličnu igru. Radnja pomalo podsjeća na film RUNNING MAN. Malog robota (vas) zatvorili su u prostoriju koja se puni vodom na veliko odusjevljenje gledalaca. Jedina šansa za spas je da se popnete do samog vrha prostorije.

Ekran je podjeljen u tri dijela. U gornjem dijelu (najmanjem) nalazi se broj bodova i života (na početku pet). U srednjem dijelu odvija se igra, a u donjem se nalaze energija, oružje i predmet koji nosite.



Nivoa ima osam i na svakom postoje dvije sakuje. Da biste se lakše uhodali u igru ukratko će vam opisati prvi nivo.

1. nivo, 1. sekcija: Ovdje možete skakati po platformama i puzati po okomitom zidu. Trebata tražiti ključeve raznih oblika i iskoristavati ih u odgovarajućim bravama, tako da nestane zid koji vam priječi put do vrha. Ključevi su skriveni u stijenama koje svijetle. Uzimate ih tako da uništite stjenju i povučete palicu prema sebi, a upotrebljavate ih držanjem pucanja. Ometate vas fantastično animirane kugle i drugi oblici. Svaki dodir s njima koštaće će vas jednog života.

1. nivo, 2. sekcija: I ovdje ćete, kao i u prvoj sekciji, skupljati ključeve. No, ovdje neće biti smetala (osim vode), već ćete morati pokupiti određen broj kristala da biste srušili zid koji vas dijeli od izlaza.

Pomagala ćete najčešće nalaziti u svjetlačim stijenama ili metalnim kutijama (ovisno o nivou). To može biti pucanje s tri smjera, pucanje gore i dolje ili nešto drugo. Najkorisnije pomagala je tzv. WATER FREEZER. Aktivirate ga držanjem pucanja, a rezultat je zamrzavanje vode. Neža lost, imate ga u ograničenim količinama.

Grafika, animacija i muzika su odlične. Scroll se odvija na dva nivoa tako da se dobija utisak trodimenzionalnosti, ali nije baš gladak.

THE KILLING GAME SHOW je odlična arkadna igra i vjerujem da će vas dugo držati uz prljavljicu.

INFO: 041/677-904 (Marko).

Heat Seeker

• arkadna avantura • C 64 • Thalamus
• 9/9

HRVOJE KARALIĆ

U svijetu otrovnih kiša jedini spas je tragač za toplinom... Planet Tantris nastanjen gnosticima i bićima iz vlažnih ogromnih prašuma, je pred uništenjem. Triphyllus Nema, tri biljke koje su srca planete, jer održavaju na životu prašume, izvor kisika na planetu, oštećene su zagađenjem koje je izazvala Industrija. Da bi spasili planet od opustošenja, gnostici su stvorili Leg (nogu), robota u obliku plave čizme sa ugrađenom loptom. Zadatak Noge je da se kroz najzagađenija područja, zaplijuskivana kiselim kišama, probije do Zeal gasnih plamenova i upije njihovu vrelinu, koju će predati Triphyllus biljkama na trećem nivou. Srce Noge je zapravo lopta: sa Fire je ispaljujete na neprijatelje, nakon čega se vraća u kiješta na sari čizme, a sa Fire i gore ispaljujete je uvis, pri čemu preuzimate kontrolu nad loptom, ostavljajući paraliziranu Nogu. Oboje neprestano skakuću. Nogu umirujete nadole, čučanj, a loptu sa Fire Razlog samostalno kretanje lopte, koja je tada bespomoćna protiv prašumskih stvorova: samo lopta može upiti toplinu Zeal plamenova. Kada je lopta u plamenu, slika lopte na kontrolnom ekranu mijenja boju od plave, crvene narandžaste do bijele lopte a žutim plamčkom u centru - to je maksimalna količina upijena toplina. Zeal plamen je tada hladan smeđi dim. Nakon upijanja topline, vratite se među kiješta na Nozi, kojoj predajete toplinu, uz hlađenje lopte i vraćanje boje. Na svakom od šest nivoa nalazi se po osam plamenova. Dok upijate toplinu ili se napunjeni toplinom vraćate Nozi, izbjegavajte prašumska bića, koja vam dodirnom oduzimaju dio upijene topline. Zato postavljajte Nogu što bliže plamenu, a bića raznesite sa pametnom bombom (dvija na nivou).

Od korisnih predmeta tu su i dvije naslage krede. Kad uđete u nju, pobijelite i zaštitite ste od kiša (koje srećom padaju samo na jednom mjestu) do sljedećeg upijanja topline.



Kontrolni ekran, osim lopte, sadrži i energiju Noge i lopte, bodova, Pass brojač te Tele ekran. Kada pokupite bombu danju, ispod njega piše Bomb, a noću Flash (bljesak). Da bi oružja djelovala, mora se ispuniti do vrha. Tele ekran vam omogućuje teleportiranje u ludu prašumu. To možete samo kada je Pass brojač ispunjen, te u njemu piše 9999. Bodove za Pass brojač dobijate ubijanjem neprijatelja i upijanjem toplina. Teleportirate se sa Shift Lock i F1.

Prašume se prostiru i preko ciglenih temelja kao sag od blata i trave, iz koje se uzdižu prekrasne, ogromne puhare sa bijelim mrljama na orvenom klobuku, niske gljive, sročliki listovi, biljke čvornatih stablika sa grozdolikim cvjetovima i ilašcem s kog kaplje voda... U tom atmosferičnom svijetu gmižu krupni puževi, mravi, paukovi i kornjače, skaču žabe, lete muhe i ostala bića. Morate dobro otvoriti oči: I najuži prolazi mogu voditi u nove dijelove prašume...

Grafika je izvanredna, likovi su krupni, bogati bojama i detaljno animirani, a zvuk je izvrstan, posebno prskanje kišnih kapi po listovima. Tehničko savršenstvo uzdiže i originalnost igre, izvedene u skrolujućoj bočnoj grafici. Igra zapravo ima tri nivoa, od kojih je svaki prikazan danju i noću (ukupno šest nivoa). Noću su boje ograničene na bijelu, žutu i smeđu, pri čemu je lopta izvor svjetlosti. Nivoi se razlikuju po biljkama i neprijateljima, a kad na trećem nivou predate toplinu trima biljkama, one će pod njenim utjecajem otvoriti latice koje će ih zaštititi od kiselnih kiša.

Lords of Chaos

● strateška igra ● C 64, spectrum ● Blade Software ● 9/9

ROMAN HORVAT

Ovo je jedna od rijetkih igara tipa »Laser Squad«. Događa se negdje oko petog stoljeća. Pojavljuju se razne kreature, spomenute u drevnim legendama. Vi ste u ulozi čarobnjaka koji se bori protiv jednog ili tri druga, tako da mogu igrati četiri igrača, čak možete stvoriti i sami svoj lik.

Na početku kupujete saveznike pomoću »mana« (imate ih 80), a možete ih i putem negdje pokupiti. Za svaki lik je potreban određen broj mana, s obzirom na njegove sposobnosti. Demoni i gigantski pauci su skupi, a patuljci jefti-



ni. Svaki lik je dobro animiran.

U glavnom meniju su ove opcije:

End move – kraj vašeg poteza; Centra – prebacujete kursor na čarobnjaka; Big map – mapa, vi ste označeni velikim kvadratom; Cast-a – kupujete čarolije; Cast-g – kupujete kreature; Pick up – uzimate predmet; Use – koristite predmet. Ovo su neki od saveznika: Goblin – mala spodobna slična Luciferu, ima puno bodova za kretanje, patuljci – nevidljivi za protivnika, Pixie – dječacić. Troll – ružno, grbavo, uhato stvorenje nalik čvarku, Centaur – čovjek-konj, Unicorn – rogati konj, Gryphon – konj-ptica, Giant – snažni div, slonovi, medvjedi, gigantski pauci, zombiji, duhovi, vampiri, medvjedi i cijeli zoološki vrt. Krećete se pomoću opcije »select-g« tako što kvadrat u nogama stavite ispred lika u pravcu u kojem želite da se kreće. Bodovi za kretanje, energija i ostali podaci su u dnu ekrana, prikazani su u crtama koje se s potezima smanjuju. Od čarolija možete kupiti čarobno oko, pomoću kojeg vidite protivnikove patuljke, vatrene kugle i druga pomagala. Neki likovi nemaju oružje, pa ga pokupite po putu. Pošto se u prvom potezu ne možete kretati, iskoristite ga za kupovanje.

Grafika je vrlo dobra, zvuk također. Nemojte se začuditi naletite li na kreature u kojima niste ni maštali. Programer je jučer ružno sanjao.

Strider II

● arkadna igra ● C 84, spectrum, CPC, ST, amiga ● Capcom/U. S. Gold ● 8/8

ROMAN HORVAT

Nastavak Stridera je bolji po grafici, zvuku i mnogo čemu. Borite protiv robota i njihovog glavnog vođe, »Super-robot«. Vaš junak još



uvijek posjeduje moć da skače oko pet metara u zrak, da radi salto od po šest metara u dalj, ali je u vremenom naučio da se vere po sasvim glatkim stjenama pod kutom od 90 stupnjeva. Kad se svemu tome dodaju bacač plamena i laser, dobili ste pravo čudovište nastalo mutacijom između King-Konga i Predatora.

Prvi nivo odvija se u šumi. Sa startne pozicije krenite desno do lifta i idite njime do vrha. Putem će vas napadati roboti i čudne leteće nakaze. Došavši liftom na određite krećete desno i najednom opazite velikog robota, koji uzaludno troši municiju na vas. Ubijte ga i krenite desno. Vratite se liftom u šumu i opet desno. Naći ćete na nešto nalik hodnicima. Verujte se i idite samo desno. Odjednom vaš se lik pretvara u robota. Već ste pomislili da je to izdaja, ali ugledali ste još većeg robota kako puca na vašeg. Odmah ste shvatili da se u robota transformišete samo kad su u pitanju jači protivnici. Ovdje je kraj prvog nivoa.

Za vrijeme učitanja vidite sliku vaše djevojke, okovane u tamnici. To vam se čini malo

čudno. Kakve koristi mogu imati od vaše djevojke? Pa oni su roboti, zar ne? Drugi nivo igrate na platformama. Idite desno i popnite se po lancu jedan kat naviše. Pazite na kugle koje vire iz stropa. Ponekad iz njih sijevne munja i oduzme vam energiju. Zatim krenite lijevo i opet po lancu gore. Idite tako do vrha zgrada, pa desno, tako da dođete na krov druge zgrade. Spustite se na dno i idite desno. Ako ste sve dobro izveli, trebali biste doći do velikog broja 4 uklesanog u zidu. Ovdje se susrećete sa tenkom. Krenite opet desno. Tu ćete često nailaziti na tankove i kuće u vietnamskom stilu, pola kuće u cigli, pola u žbuci. Nakon tankova evo opet vas u obliku robota. Borite se protiv nekog helikoptera. Ako vam u borbi nestane energije, opet ste Strider.

Na trećem nivou muvajte se pretežno nadole dok se ne pretvorite u robota i započnete bitku u ogromnim čudovištem, koje kleči tako da vas može pogoditi.

Četvrti nivo odvija se na nebu. Penjite se koliko možete i ubijte glavu koja bljuje vatru, pomicaluđi se tamo-amo.

Peti nivo je i zadnji. Djevojku koju ste gledali pri učitanju, sada pobjednički nosite.

Ovo je dobra pucačina u kojoj treba malo i misliti. Sve je donekle dobro urađeno, a ideja u spašavanju otete djevojke je potpuno otrcana.

Monty Python's Flying Circus

● arkadna avantura ● C 64, spectrum, CPC, ST, amiga, PC ● Virgin Mastertronic ● 7/8

HRVOJE KARALIĆ

John Cleese, Michael Palin, Terry Gilliam, Terry Jones, Eric Idle i Graham Chapman svojim su neobičnim i surovim humorom proslavili britansku seriju Leteci cirkus Monty Pythona, koja je prikazana i kod nas.

Gumby, karikaturna crtana tvorevina Palina, je izgubio mozak, koji je raskomadano u četiri dijela, na svakom nivou po jedan. Komad mozga se ne može pokupiti: ako završite nivo pokupivši 16 plavih konzervi, dobivate komad mozga (Brain piece award). Konzerve dobijate ubijanjem većih neprijatelja i uništavanjem naslage cigli. Vaše oružje su ribice kratkog dometa u ograničenom broju.

Nivoi se sastoje od manjih podnivoa, pa se Gumby pojavljuje u četiri oblika: karikaturna ljudeskara, ptica s Gumbyjevom glavom, Gumby-riba, te glava nasadena na noge na gumby nivou. Na bonus nivou ne pucate, već skačete po ciglama koje se poslije skoka lome, da pokupite voće i konzervu.

Gumbyja gledate u bočnoj skrolovanju. Push up skroling ima manu da se ekran pomiče udesno kad dođete do njegovog kraja, pa ćete previdjeti neke konzerve. Srećom, pauzom sa Space i Run stopom vraćate se na početak, gdje pritiškom na Space birate između kratke i loše muzike i dobrog zvuka, a s Fire započnete igru. Grafika je osrednja, ali je podližu smiješni sprajlovi, a pozadina dijelom preslikava špicu serije.

U obliku krupnog Gumby-čovjeka krećete se crvenim tlom iz koga rastu ruke i noge u daljnji praćeni hodajućim oblacićem. Opasnost su tegovi koji padaju s neba. Na ulazu u cjevovod nalazite se u krijušili G-ribe, koja pliva u vodnom ispunjenim cijevima u mraku, uz potmull podvodni zvuk ubijajućih ribe i noge s propelerima. Velika cijev se na dnu račva na marje cijevi



kojih naviru neprijatelji, ili koje vas vode u druge dijelove cjevovoda. Na nekim cjevčicama nasađene su trube. Kada ih pogodite, dobivate razne pogodnosti: biće koje ostavlja pametnu bombu, silu koja odljepljuje konzerve na vrhu, ili silu koja diže bodiljkavi poklopac koji blokirao prolaz cijevi. Ona vas vodi do plemića sa propelerom na glavi, koji vas gađa u zelenoj silici. Zatim dolazite do podvodne mreže cijevi, gdje plutaju televizori, dva gangstera, policajac u dječjem gumenom pojasu. Nakon policajca postajete »normalni« Gumbi, i završavate nivo nakon ubijanja letjećih nogu.

2. U ulozu Gumbi-čovjeka krećete se tlom iz koga rastu narandžasti stupovi i ogromne noge sa stopalom uvis. Pucanjem u velike usne na stupovima, sa neba se odljepljuju konzerve. Napadaju vas engleski džentlmeni koji skakuću na jednoj nozi. Tu i savitljive gume koje vas bacaju uvis. Pucajte u usne da se crna rupa spustila u ružičastog neba i teleportirala vas na ogromne noge, pored eksplozivnih mačića. Pažljivim pucanjem u cigle napravite stepenice do konzerve u visini. Došli ste do lebdeće noge sa usnama ispred kojih lebde cigle na lijevoj strani, platformom na sredini ekrana, dok se gore desno nalazi top ispred koga leti biće na balonu pucajući na vas. Ispod topa, dolje desno, leži rupa. Lijevo od rupe je stup iznad koga lebde četiri cigle. Uništite ih, osim cigle dolje desno. Sa stupa skočite na nju, zatim lijevo na platformu u kojoj ćete udariti lopticu, te pucajte u usne. Topa će izletjeti tane, ubivši balon iz koga na to pada konzerva. Udite u rupu, i naći ćete se u svijetu sazdanom od ljudskih kostiju, između kojih umjetne čeljusti pucaju na vas. Pazite koju ciglu prvo uništavate, jer morate pravilnim gađanjem praviti stepenice do konzerve u visini. Ometaju vas skakutajuće glave, a rupu spuštate s neba pucajem u crnu ribu među kostima. Nalazite se ispod glava, obješenih za ljubičaste liane, među stupovima koji završavaju šakama. Zatim dolazite do stupova i nacrtanim licima, bijelih rukavica koje poput pauka gmižu na pratima, i upaljenih bombi. Slijede stupovi na kojima skače ogromna puščica koja ispaljuje vrh svoje njuške kao raketu.

3. U obliku pilce letite među oblacima. Ulaškom u crnu rupu letite ispod čepelina, iz čijih rupa padaju tegovi, a neke vode na bonus nivoe s nebom i američkom zastavom kao pozadinom. Crna rupa vodi vas u svemir, gdje između planeta lete humanoidna čudovišta i krilati konji pegazi. Na nebu iz velikog frižidera ispadaju bradali patuljci, a zatim vas napadaju leteće rukavice i bića na prozorima. Na kraju vas napada smiješna, velika kokoš s kaubojskim šeširom koja nese lebdeća jaja s cilindrima.

4. Kao Gumbi-čovjek nalazite se na pokretnoj traci i tvornici. Zavirite u svaki tunel i prolaz jer se na bonus nivoima kriju konzerve! Pucajte u cijev s crnim otvorom, i konzerva iznad vas se odljepljuje. Ometaju vas crne lopte i pokretni stupovi. Iznad druga lopte nalaze se cigle, koje vam pravilno složenje omogućuju prolaz do cijevi u visini koja odljepljuje konzervu. Konzerva pada na traku koja je baca u bezdan. Što prije uhvatite konzervu! Dolazite do prolaza nadolje ispred koga se otvorena metalna ruka u topov-

skoj cijevi ispaljuje u obliku šaka. Zatim ubijate glavu engleskog gospodina nasađenu na nojliku spravu. Sa trake za proizvodnju na vas jure krupni auti igračke, dok sa elastične gume skačete uvis uz kombinacije za dobivanje konzervi. U hali i zavrtnjima i zupčanicima napada vas crna pingvinska maska nasađena na klupko vune. Ispaljujući metke i sjekire iz kljuna. Nastavljate napadati neuništivim zečicama i mjehurastim kreaturama, do vojničke glave. Vojnička kapa sa propelerom odvaja se od glave čim se otvora duga usta. Slijedi iznenađenje: mračna srednjovjekovna ođaja s plamtačim bakljama na kamenom zidu po kojoj leti kralj crvenom prijestolju pucajući na vas.

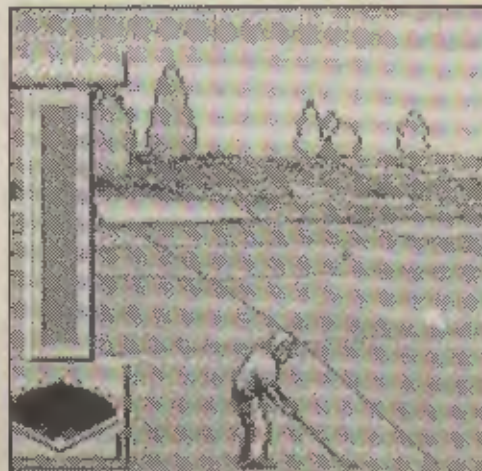
Ako ste pokupili sve konzerve, slijedi po programerima »spektakularna završna sekvenca« koju otkrijte sami.

Greg Norman's Ultimate Golf

● sportska simulacija ● C 64, ST ● amiga, PC
● Gremlin ● 8/8

MITJA MLADKOVIĆ

Dugo smo čekali na pravog nasljednika izvanrednog Leaderboarda. Greg Norman's Ultimate Golf veoma mu se približio. Ekran je podjeljen na dva dela. U donjem, gde se odvija igra, nalazi se mnogo različitih podataka. Gornji deo ekrana je rezervisan za mnogo različitih podataka. Gornji deo ekrana je rezervisan za mnogo opcija: CLUB – možete da birate jednu od sedamnaest palica; SWING – dobitete podatke o smeru vetra, a možete da podesite i zamah kod udarca (swing), položaj palice kod udarca (side) i visinu u kojoj ćete udariti lopticu, MAP – ovom opcijom posmatrate teren iz ptičije perspektive i šetate njime; INFO – dobijate podatke o snazi vetra, broju udaraca... Tu su još podopcije: Change hole (menjaš rupu), Replay hole, Replay shot i Move; PLAY – levo birate snegu i ugađ udarca, a ostalo obavlja računar; SOUND – objašnjenja su suvišna; EXIT – na ekranu se pojavljuju podopcije Save game, Continue (nastavak igre), Restart (opet na početak) i Quit (kraj programa). Grafika je odlična, a novost je ta da teren nije ravan kao kod većine simulacija, već talasasi. Posle svakog zadatka računar ocenjuje i komentariše vašu igru. Nedostatak Ultimate Golfa je, možda, samo to što u računar treba učitati svaku rupu posebno.



Sly Spy – Secret Agent

● arkadna igra ● amiga, spectrum, C 64, CPC, ST, PC ● Ocean ● 8/8

GRGA MRKONJIĆ

Igra nije konverzija s automata, niti je rađena po filmu. Iako ima velike sličnosti s filmovima o Jamesu Bondu. Sly Spy je agent pred kojim nema tajni. Upišite svoj tajni kod i igra može početi.

Išaćete iz avlona. Sa svih strana dolaze neprijatelji. Ovaj nivo nije osobito težak. Na kraju otvarate padobran s američkom zastavom, a u pozadini vidite velegrad.

Spremate padobran, uzimate pucajku i krećete. Već pri prvom koraku napada vas šestica ljudi. Nakon što završite ovaj dio igre, pojavljuju se mapa, zgodna sličica i opis slijedećeg dijela. Sjedate u motor u želji da sustignete crni sedan u kojem su okorjeli kriminalci. Ometaju vas motoristi i kriminalci iz zraka. Uništivši sedan krećete dalje. Ponovo se pojavljuju sličica, mapa i opis slijedećeg dijela igre. Nivoa je osam, ali ih otkrijte sami.

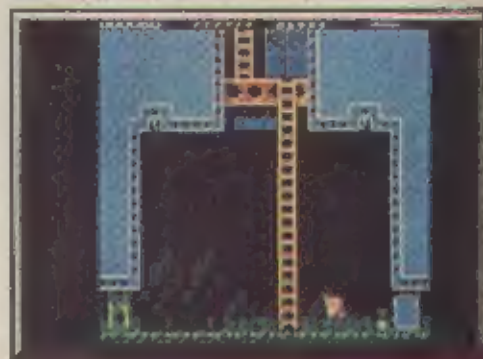
Iako igra zauzima dvije diskete (verzija za amigu), preporučujem da je uvrstite u vašu kolekciju, zbog dopadljive grafike i koliko-toliko originalne ideje.

Rick Dangerous II

● arkadna avantura ● C 64, ST, amiga, PC
● MicroStyle ● 9/9

MARIN MARUŠIĆ

Rick se vratio iz svoje avanture po Amazoni i Egiptu, i baš se opustio i otpočeo miran život kada je dobio pismo upozorenja od »debeljka« (fat man) koji hoće da uz pomoć tehnički nadmoćnijih svemiraca zauzme zemlju. Pošto vl (Rick) to načete dopustiti, ostavljate svoju dragu



u restoranu i trčite u Hyde park gdje vas očekuje svemirski brod.

Ovo bi bio kratak i pojednostavljen zaplet Ricka II. Što se tiče ideje, i programeri su malo zabrijali, jer su igrači alergični na svemirce koji napadaju zemlju. Ali izvedba je stvarno duhovita i maštovita, a grafika, animacija i zvučni efekti su veoma dobro urađeni. Igra važe igračka kompjuter i potiče na logično razmišljanje, pri čemu su potrebni dobri refleksi.

Prvi nivo je svemirski brod kojeg morate oteti. Nivo nije baš jednostavan, ali poslužit će kao trening za iduće, mnogo teže. Kao i uvijek, tu su razne klopke, roboti, topovi, laseri, lilitovi i ostala čuda tehnike koja vam otežavaju život. Prepreke možete isključiti pritiskom određenog,

pazite, određenog prekidača. Robote otklanjate sa fire + gora, dok bombe koristite protiv pregrada koje vam blokiraju prolaz (fire + dolje).

Drugi nivo se odigrava na ledenoj planeti punoj još maštovitijih i težih prepreka. Ako i ovaj nivo pređete, što je vrlo, vrlo teško, eto vas u prašumi na sasvim drugoj planeti. Tu vas očekuju razne kreature koje jure za vama, odroni, kugle što se kotrljaju sa vrha ekrana, strelice koje lete prostorijama, skriveni bodeži koji vam dolta mnogo otežavaju put.

Ako i ovaj nivo pređete (u što sumnjam, ako nemate poklice), dolazite do rudnika koji je težak, pretažak. Tu su sada najraznovrsnije, najgora, najduhovitija, najbrojnije klopke koje smo susreli u igrama silčne vrste (bolje da ne nabrajam). Poslije ovog dolazi do obračuna između svemiraca na čelu sa »debeljkom« i vas.

Imperium

• strateška igra • ST, smiga • Electronic Arts • 9/9

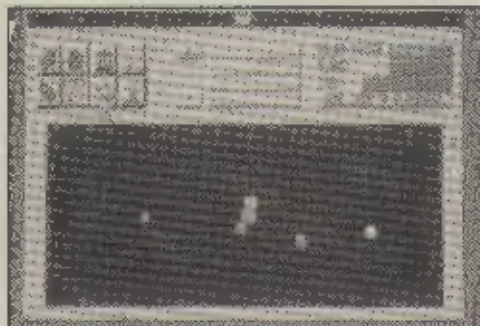
SINIŠA KONJEVIĆ

Imperium vas stavlja u ulogu novozabranog vladara imperije Thoygon (koja obuhvata Zemlju, Mars, Veneru, Saturn i Jupiter), sa zadatkom da Imperiju proširujete, branite od raznoraznih osvajača i nakon pedeset godina vladavine ponovo budete izabrani za vladara.

Vladati možete potpuno samostalno. Ili uz pomoć kompjutera koji će neka formalnosti obavljati umjesto vas. Iako se upravlja uz pomoć

sisstema ikona i enija, za one koji se prvi put susreću sa igrom, igranje će predstavljati pravu moru. Zato slijedi kratko uputstvo.

Nakon što odredite vaše ime i ime imperije kojom ćete vladati, igra može da počne. Sada možete da odredite u čemu će vam kompjuter pomagati (vojska, ekonomija, diplomatija) ili jednostavno da sami krenete u akciju. Ovdje bih



vam savjetovao da obavezno isključite muziku, jer oduzima prilično vremena dok se učita i dekomprimuje. Sada nekoj imperiji ponudite ugovor o saradnji (Alliance Construction menu). Ako ponudite zadovoljavajuće uslove (dovoljno novca, te dobre trgovačke i poslovne veze), dobijete saveznika koji će vam mnogo olakšati osvajanje svemira i očuvanje vaših granica. Prebacite se u Treasure meni i opcijom Transfer Health prebacite novac na račun federacije.

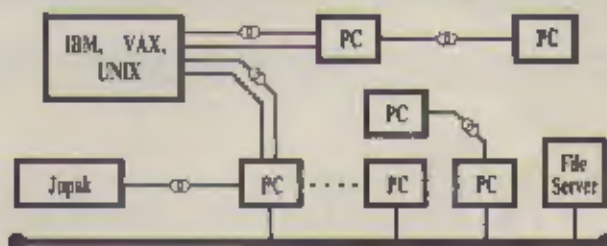
Sada možete otpočeti gradnju vlastite flote. Za početak izgradite nekoliko brodova čiji su planovi gotovi, jer posjeduju veliku brzinu neophodnu za osvajanje svemirskih prostiranja. Kompaniji koja proizvodi brodove dostavite no-

vac neophodan za njihovu izgradnju (opcijom Ship Build Status u Military meniju). U međuvremenu oformite vlastite trupe na svim planetama koje posjedujete (Create Troops), ali pazite da svi rodovi budu podjednako zastupljeni. Poslije ove akcije odredite poreze i carine na robu koju izvozite odnosno uvozite (Taxation meni). Za uspješnu vladavinu neophodan je i dobar odnos sa potčinjenima. Nagradite ih titulama, činovima i dobrim platama (Subordinate Display), ali pazite na njihove sposobnosti i što je najvažnije, njihovu odanost prema vama (ako nekog izostavite, postaćete vam ljut protivnik na izborima). Pogledajte kako stojite sa uvozom i izvozom artikala (meni Treasure - Imperial Volume of Trade). Ako vam se neki odnos učini ekstremno nepovoljan, možete da zatvorite granice za tu imperiju (Embargo meni). Nakon svega priložite nekoliko hiljada kredita za svoju kampanju. Završite tekuću godinu izborom opcije Next Game Turn.

Pošto pogledate poruke pristigle u međuvremenu i saznate da su vam brodovi gotovi, formirajte vlastitu flotu (Military meni, opcija Create Fleet). Odredite komandira, te broj vojnika i oprema koju želite da ukrcate. Sada popunite trupe na svim planetama koje držite. Pogledajte mapu i nađite najbližu planetu koja ne pripada ni jednoj imperiji. Floti dajte naredbu da okupira planetu. U međuvremenu razvijajte brod sa jakim naoružanjem i oklopom koji će braniti centar imperije (Zemlju). Prekontrolišite poreze, carine, proizvodnju i vjernost vaših podanika. Ako je sve u redu, povucite slijedeći potez. Dalje ćete moći i sami.

Igra se završava kada neko zauzme Zemlju ili kada ne pobijedite u izborima. Imperium vas neće ostaviti dok ne uništite protivničke imperije.

Kad poznamo sva slova abecede, možemo da pišemo.



Novell lokalna mreža

- lokalna veza
- (o) — veza preko modema

Pojedini računari su kao slova abecede. Tek kad ih povežemo, možemo da sastavimo reči i rečenice.

Mi kod LANComa u tome možemo da vam pomognemo.

Profesionalno i bavimo postavljanjem informacionih sistema na osnovu Novell lokalne mreže.

- savetujemo i projektujemo
- dobavljamo i instaliramo kompletnu računarsku i aplikacionu programsku opremu
- uvodimo vas u rad i školujemo
- održavamo svu opremu i tako omogućavamo vaše neometano poslovanje
- reference: preko 50 lokalnih mreža godišnje u svim većim mestima Slovenije i u većim gradovima Hrvatske
- tražimo poslovne partnere po Jugoslaviji - pozovite nas!

LANCom
INŽENIRING

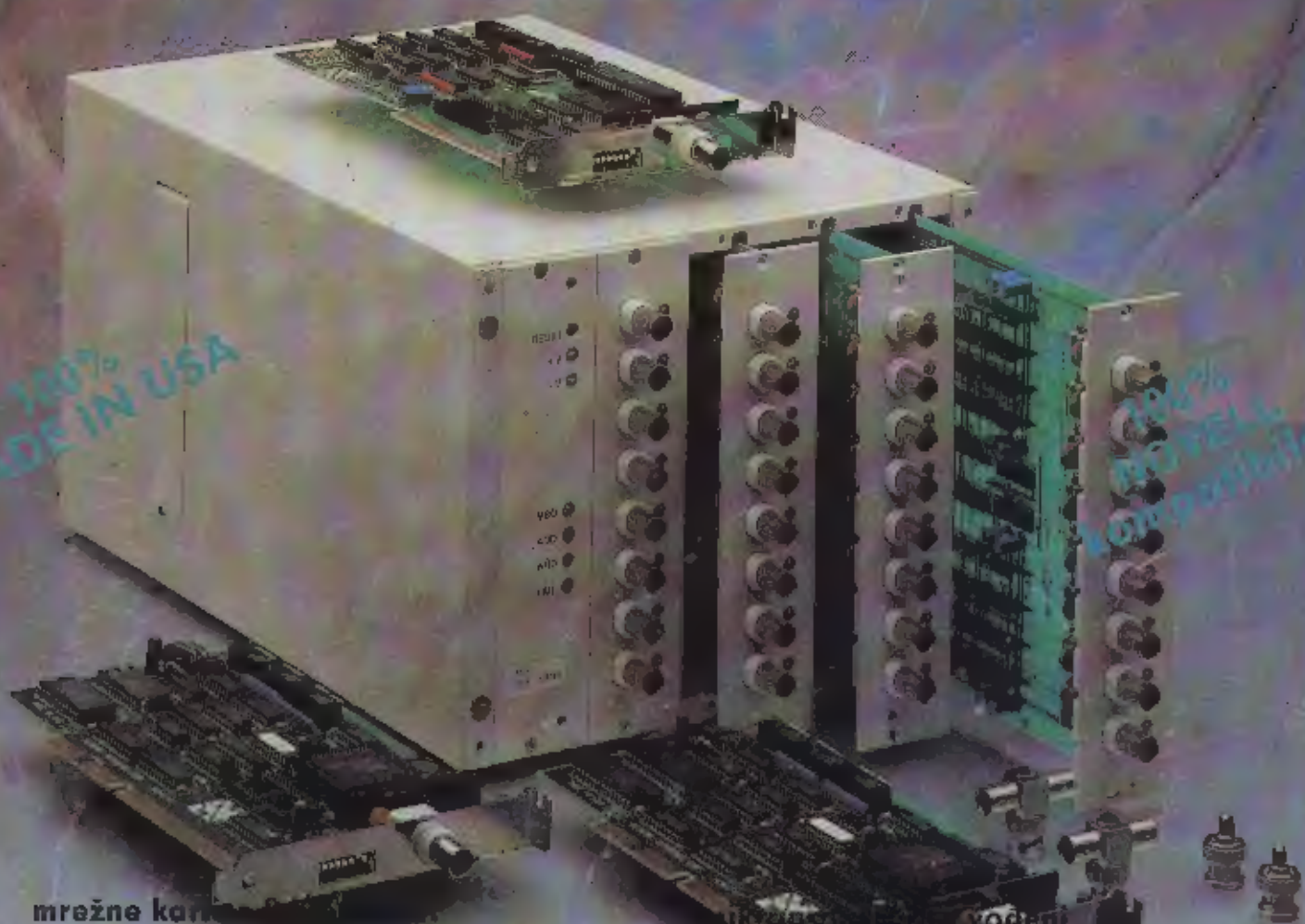
62000 Maribor, Parizanska 3-5, tel. (062) 222-826, 211-061, 29-061, faks: (062) 27-684

LANCOM

potpuna paleta mrežnih proizvoda ARCNET NETWORK INTERFACE CORP.

100%
MADE IN USA

100%
NOVELL
SUPPORTING



mrežne kartice

- potpuna kompatibilnost sa SMC/RxNET/Pure Data
- dijagnostička LED vidljiva sa zadnje strane
- podešavanje naslova (node ID) sa zadnje strane
- I/O memory prekidači vidljivi i pristupačni, kad je kartica montirana u PC
- upotreba 16 K memorijskog prostora ili manje
- mogućnost upotrebe u TURBO načinu
- opcija 8 K RAM buffera
- buffer chaining
- circular Buffering
- bez dodatnih stanja čakanja
- bez mostića (jumpera)
- svaka kartica »funkcionalno testirana«
- 500% brža t.zv. »arbitration speed« od standarda
- 11 mogućnosti prekida

aktivni mrežni proizvodi INTELIGENTNI AKTIVNI HUB:

- potpuna kompatibilnost sa svim mrežnim greškama
- automatsko preklapanje između matičnih grana
- automatsko preklapanje između naslova
- automatsko pasivno uključivanje, kad je problem eliminisan
- upotreba postojeće instalacije
- podržava redundantne parove kablova
- software teče iz bilo koje radne stanice u mreži
- vođenje statistike mreže i svih grana
- ne duplira probleme od hub-a do hub-a
- minimalni mrežni overhead
- stablo priključnih grana sa imenima tekstova
- vođenje liste događaja
- upis vremena za poslednje 64 rekonfiguracije
- više hub-ova može da deli jedan naslov (node ID)
- udružljiv sa svim grafičkim adapterima

uskoro kompletna paleta mrežnih proizvoda ETHERNET

ARCNET mrežni produkti su na raspolaganju u koaksialnoj, twisted pair, single fiber optic i double fiber optic. Nudimo vam takođe pripadajuće kablove, BNC konektore svih vrsta, T-članke, zaključne članke (terminatore), sva proširenja HUB, pasivne HUB...

ARCNET je zaštićena robna
znamka DATAPOINT Corporation
NOVELL je zaštićena robna marka
NOVELL inc.

 **MARAND**

Inženiring, 61000 Ljubljana, Kardeljeva plaščod 24

Tel. (061) 340-652

(061) 371-114

Fax. (061) 342-757